MCR-74-423 Rev.1
NAS1-14450

NASA CR-132743

Volume III

Final Report

March 1977

Programmer's Manual

Six-Degree-of-Freedom Program to Optimize Simulated Trajectories (6D POST)

(NASA-CR-132743) SIX-DEGREE-OF-FREEDOM PROGRAM TO OPTIMIZE SIMULATED TRAJECTORIES (6D POST). VOLUME 3: PROGRAMMER'S MANUAL Final Report (Martin Marietta Corp.) 231 p

N77-76688

Unclas
00/61 21658

APR 1977
RECEIVED
NASA STI FACILITY
INPUT BRANCH

MARTIN MARIETTA

NASA CR-132743

MCR-74-423 Rev. 1
NAS1-14450

Volume III

Final
Report

March 1977

PROGRAMMER'S MANUAL

SIX-DEGREE-OF-FREEDOM PROGRAM TO OPTIMIZE SIMULATED TRAJECTORIES (6D POST)

SUPERCEDES NASA CR-132743
November 1976

Prepared by

G. L. Brauer
and R. Stevenson

Approved

D E Cornick
D. E. Cornick
Program Manager

MARTIN MARIETTA CORPORATION
P.O. BOX 179
Denver, Colorado 80201

NASA CR-132743

MCR-74-423 Rev. 1
NAS1-14450

Volume III

Final
Report

March 1977

PROGRAMMER'S MANUAL

SIX-DEGREE-OF-FREEDOM PROGRAM TO OPTIMIZE SIMULATED TRAJECTORIES (6D POST)

SUPERCEDES NASA CR-132743
November 1976

Prepared by

G. L. Brauer
and R. Stevenson

Approved

D E Cornick
D. E. Cornick
Program Manager

MARTIN MARIETTA CORPORATION
P.O. BOX 179
Denver, Colorado 80201

FOREWORD

This final report describing the formulation of the Six-Degree-of-Freedom Program to Optimize Simulated Trajectories (6D POST) is provided in accordance with Part 3.0 of NASA Contract NAS1-14450. The report is presented in three volumes as follows:

Volume I - 6D POST - Formulation Manual;

Volume II - 6D POST - Utilization Manual;

Volume III - 6D POST - Programmer's Manual.

This work was conducted under the direction of Mr. Richard Powell of the Space Systems Division, National Aeronautics and Space Administration, Langley Research Center.

CONTENTS

		Page
	SUMMARY	iv
I.	INTRODUCTION	I-1
II.	PROGRAM STRUCTURE AND LOGIC	II-1
	Overall Program Logic	II-2
	Outline of Program Logic	II-5
	Tape or File Designation	II-6
	Program Additions	II-9
	Adding New Auxiliary Calculations	II-13 and II-14
III.	Subroutine Descriptions, Flow Charts, and Selected Listings	III-1 thru III-181
IV.	Service Routines	IV-1 thru IV-5
V.	Definition of Internal FORTRAN Symbols	V-1 thru V-25

Figure		
II-1	Program Macrologic	II-3

Table		
II-1	Summary of POST Memory Requirements	II-1
III-1	Contents of Data Buffers	III-141

SUMMARY

This volume documents the program logic, subroutine descriptions, and other information concerning the Six-Degree-of-Freedom Program to Optimize Simulated Trajectories (6D POST) of interest to the programmer.

I. INTRODUCTION

The program was written according to guidelines designed to provide complete generality wherever possible, using a minimum of central processor time and memory. The guidelines adhered to are:

1) Variable computer memory requirements dependent upon the problem, with an absolute minimum allocation of 66025 octal words.

2) CDC Fortran IV extended and CDC compass version 3.2-406 programming languages. Assembly language programmed routines to reduce central processing time.

3) Minimum program execution code.

4) Modular program organization.

5) Generalized routines to permit simulation of various types of vehicles.

6) Generality of input, output, targeting and interrputing variables.

7) Compatibility of operation on CDC 6000 series computers using either Scope 3.4.3 or NOS 1.1 operating systems.

II. PROGRAM STRUCTURE AND LOGIC

6D-POST is coded in CDC Fortran IV extended and CDC compass version 3.2-406 assembly language. An overlay structure is used to minimize program memory requirements and disk storage is used to minimize the memory required for table data. The program requires a minimum of 70,000 octal words of computer memory. The minimum field length requirement for any particular problem is dependent upon the amount of table data associated with the problem. The amount of table data stored on disk is a user-controlled variable, execution time can be reduced at the expense of using more memory and vice versa.

Executive programs are used throughout. These control the program flow by invoking routines which contain the actual mathematical formulations. This structure allows the program to be modified quickly and easily.

All executive programs and load time defining data routines are maintained on file POST-6D. All other nonsystem routines are maintained on either the UTILIB, OPTIM or POST-6DL libraries.

TABLE II-1
SUMMARY OF POST MEMORY REQUIREMENTS

Overlay	Basic Program	Scope 3.4.3 System	Subtotal (Absolute)
(0,0)	12310	15177	27507
(1,0)	31133	4417	35552
(2,0)	14452	1144	15616
(2,1)	6127	--	6127
(2,2)	12127	576	12725
(2,3)	16420	547	17167
(2,4)	706	66	774
(2,5)	10545	14	10561
(2,6)	6020	71	6111
Operating total (minimum) is 66025			

Overall Program Logic

6D-POST is structured in three overlay levels, as shown in Figure II-1. The first overlay (0,0) is the master executive overlay, which controls the overall program. This overlay controls the read-in or input data and determines which trajectory computations are to be performed.

Overlay (0,0) first calls overlay (1,0), which reads the namelist input data from cards and stores the processed data on disc for later use.

Overlay (2,0) is called by (0,0) after (1,0) has completed the input processing tasks. The first decision in overlay (2,0) concerns the type of simulation; i.e., single trajectory or search/optimization mode. If a single trajectory is to be run, the program calls overlays (2,1), (2,2) and (2,3) sequentially, then returns to the master overlay (0,0). If the search/optimization mode is to be used, the program control is turned over to subroutine MINMYS, which calls overlays (2,1), (2,2), (2,3), (2,4), (2,5) and (2,6) as required to perform the search/optimization function. When convergence has been achieved or the maximum number of iterations has been exceeded, control reverts back to the master overlay (0,0) for the next problem.

An outline of the approximate calling sequence for each routine is presented in the following section of this report. This outline shows which subroutines are called by a given routine, thereby allowing the detailed logic flow to be followed easily. The overall program logic described by the overlays is as follows:

1) Overlay (2,1) reads the previously processed input data from tape, locates the data for the current phase (event) and initializes the program values based on this input;

2) Overlay (2,2) initializes the equations of motion for the current phase;

3) Overlay (2,3) integrates the equations of motion from time t_i to a specified stopping condition for the current phase;

4) Overlay (2,4) initializes the targeting and optimization variables for the current problems;

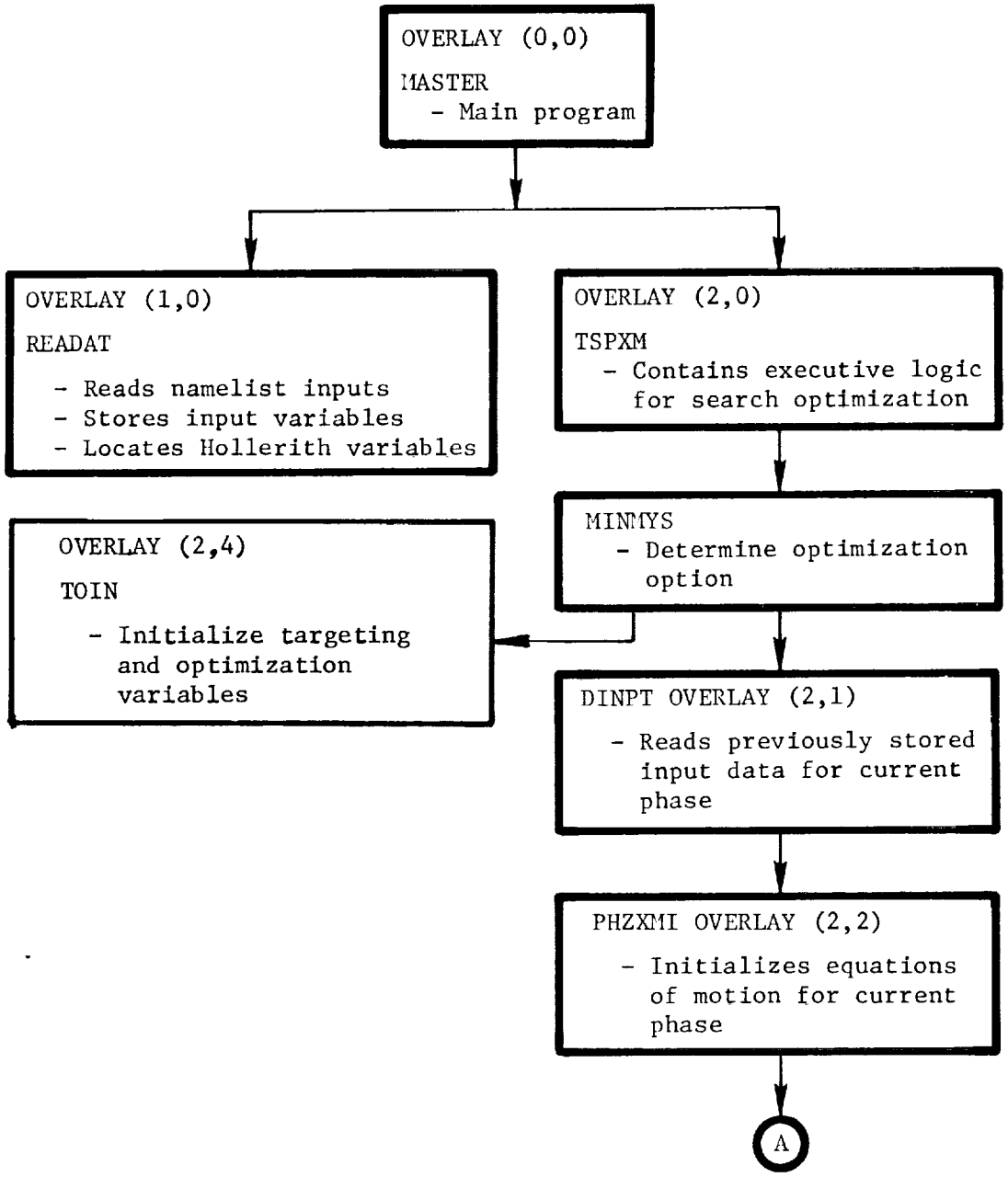


Figure II-1. - Program Macrologic

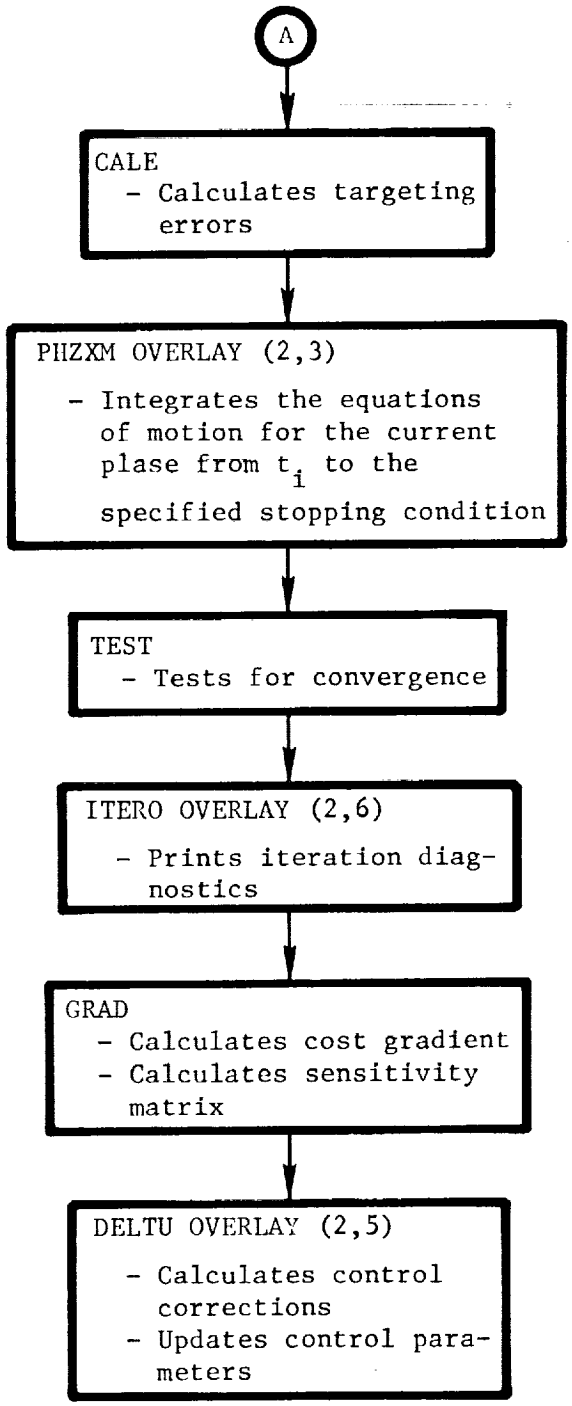


Figure II-1.- Concluded

5) Overlay (2,5) calculates the control corrections based on the search/optimization algorithm being used, limits the control parameters that violate the control parameter constraints, and tests for convergence;

6) Overlay (2,6) prints out an iteration summary at the end of each iteration. It also performs any other information output tasks required by search/optimization algorithm, such as printing trial step summaries.

The program dictionary (subroutine DICT) performs a one-to-one mapping of variables in common and the Hollerith names by which the user can select the variables for a variety of uses, including output, stopping conditions, control variables, and targeting variables.

All variables in the dictionary are located in common with respect to two labeled commons, IV and END. The first of these commons defines the starting reference; the last defines the ending reference. These commons must bracket all commons required by the dictionary.

6D-POST uses a generalized table storing and look-up procedure. Each table has its own multiplier, which is implemented by dimensioning the table by (2). The first location contains the address of the table and the second location contains the table function multiplier. Tables are stored on disk to minimize memory requirements and are transferred from disk to memory only when required. The number of tables resident in memory is variable and can be modified through the input. The generalized table lookup (GENTAB) is set up to handle all allowable types of tables, namely, constant-value, monovariant, bivariant, and trivariant.

Outline of Program Logic

This outline shows the calling sequence for a single iteration for the trajectory and optimization logic. Certain routines are called only if certain options are requested. These routines are presented in their approximate calling sequence. The outline allows the user to follow the program flow either forward or backward from a given routine to aid in understanding the logic flow. Each subroutine that is called by another routine is listed immediately below and to the right of the calling routine.

Tape or File Designations

The program uses several Tape (File) designations internally to perform the simulation tasks. These files are normally stored on discs, but tapes can be used by assigning them to proper file designations. The file designations are as follows:

Tape (or File)	Definition
1	Contains the general data and table multipliers for the problem
2	Contains the initial conditions for each event that has a control parameter
3 and 4	Store input data for multiple runs
5 (INPUT)	Stores input data
6 (OUTPUT)	Stores output data
8 (PROFILE)	Contains the simulation profile

Common Designations

POST uses several labeled commons to provide communication between subroutines. In addition, a blank common is used to act as a data buffer for the table input data and the event criteria. The blank common could be labeled, if desired, without adversely affecting the operation of the program.

The labeled commons are briefly described below in alphabetical order. The variables are listed in the following section alphabetically to provide an easy cross-reference.

AUXVC: Common AUXVC contains the variables that are computed as auxiliaries at the end of each integration step.

CYCVC: Common CYCVC contains variables and flags used to perform cycling functions during forward integration.

DPGVC: Common DPGVC contains the variables and flag associated with the guidance (steering) options.

DYNVC: Common DYNVC contains variables and flags required to perform dynamics functions during the forward integration. Primarily this includes time references and discontinuity flags.

DYTEM: Common DYTEM contains variables and storage used by the integration algorithms to integrate the equations of motion forward. No variable in this common may be input or output.

END: Common END is used to define the end of the dictionary. Any variable defined in a common after common END cannot be input, output, or used as a search parameter. This common contains only one variable, namely, END.

GENIC: Common GENIC contains variables of a general nature that are required in overlay (0,0).

GUIDIC: Common GUIDIC contains the input variables for the generalized guidance routines.

GUIDVC: Common GUIDVC contains the computed variables for the generalized guidance routines.

HØLINC: Common HØLINC contains all of the Hollerith input variables.

INFVC: Common INFVC contains variables and flags that may be used in the information output routines at any phase.

IV: Common IV is used to define a reference to the dictionary region. All variables that are to be input, output, or used as search parameters must be defined in a common between common IV and common END. IV contains the size of this region. This common contains variable, namely, IV(2).

LØCAL: Common LØCAL contains parameters used in computing the equations of motion and the auxiliary equations that are not required to be input or output. If a common variable is to be added and it is not needed as an output or an input, it should be added to this common.

MNMMLT: Common MNMMLT contains a list of mneomonic multipliers associated with the aerodynamic tables. The first cell contains the value 1.0. The remaining cells contain the address of any input variable within the dictionary.

MØTBL: Common MØTBL defines all tables to be interpolated by the general table lookup routine GENTAB. Each table requires two consecutive storage locations. The first is the table address and the second is the value of the table multiplier. Whenever a table is added to this common, subroutines DICT and DATA must be modified accordingly.

MØTIC: Common MØTIC contains all parameters that are required as input to the equations of motion. Input parameters do not have to be defined in this common; however, when such a parameter is defined in a lower common (e.g., MØTVC), the program must search for the dictionary and, hence, run longer.

MØTVC: Common MØTVC contains all variables used in the equations of motion. These are generally not input or constant parameters. They are available for output, table arguments, or search parameters through the dictionary.

MULTRC: Common MULTRC contains the variable associated with the multiple-run capability.

ØVRLY25: Common ØVRLY25 contains the variables required by overlay (2,5), which contains the direction-of-search logic.

PHZVC: Common PHZVC contains flags and constants required to perform the phasing functions.

REDAT: Common REDAT is defined in overlay (1,0) by READAT and contains variables and storage data required to build the general and table data buffers.

SEARC: Common SEARC is defined in BLKDAT and, in general, contains all parameters required by the iteration algorithms. Variables in common SEARC can be input only once per run through namelist SEARCH. They cannot be changed through input at a phase. Since SEARC is defined in overlay (0,0), it is available to every routine in the program.

SERVC: Common SERVC is a service common available to all routines in the program. This common contains 50 cells of temporary storage, 5 commonly used index parameters, and a list of the most frequently used fixed- and floating-point constants. This common should be used whenever possible in order to conserve storage.

SPECAL: Common SPECAL contains the variables associated with the special calculation routine CALSPEC.

TARGVC: Common TARGVC contains parameters calculated for the target vehicle.

TGOVC: Common TGOVC contains variables and flags required to perform the time-to-go functions.

Program Additions

The guidance, navigation, and flight control routines will generally be coded by the user. The coding of these routines may require the user to make minor program additions. The most frequently requested types of program additions are: addition of new general variables, addition of generally new tables, and addition of new integrals. Instructions for making these additions are presented in this section. Other types of additions will generally require in-depth knowledge of the program code, and a programmer familiar with the program should be consulted.

Addition of new general variables.- General variables are any variable that are computed in the simulation portion of the program and are to be input, output, used as table arguments, search parameters, integrals, or derivatives. The program Executive processing algorithm expects to find all general variables defined in a labeled common which is loaded between the labeled common /IV/ and the labeled common /END/. The labeled commons /IV/ and /END/ are defined in subroutine DICT for overlay (1,0) and in subroutine DATA for overlay (2.0). The labeled commons defined between /IV/ and /END/ must be in the same order and must be the same size in subroutine DICT [overlay (1,0)] and in subroutine DATA [overlay (2,0)]. That is, a one-to-one mapping of parameters in subroutine DICT to subroutine DATA relative to labeled common /IV/ must be maintained; /END/-/IV/ in overlay (1,0) must equal /END/-IV/ in overlay (2,0). This is absolutely required for proper program operation.

The following steps should be followed to add a general variable:

1) Add the new variable(s) to an appropriate labeled common. For example, if the new variable is an auxiliary parameter, it should be added to labeled common /AUXVC/. New variables should be added on to the end of an existing common. Only the length of the common should change, NOT the structure.

   If the user does not want to add to an existing common, a new labeled common may be defined, and the new variables included in it. However, this is not generally necessary.

2) The labeled common to which the new variables have been added, or the new labeled common, is replaced or added into subroutine DICT in overlay (1,0). If a new labeled common is being added it must be placed after common /IV/ and before common /END/; but not between common /MOTBL/ and common /MOTEND/. The locations from common /MOTBL/ through common /MOTEND/ are reserved for tables.

3) For every new variable added, the Hollerith name by which it is to be known must be set into its location for use during input processing. This is done by a DATA statement in subroutine DICT. For example, if a new variable called AROANG is added, then the data statement DATA AROANG/6HAROANG/ must be added in the subroutine DICT.

4) If the new variable is going to be on input quantity it must be added to NAMELIST/GENDAT/. The input NAMELIST/GENDAT/ is defined in subroutine RGENDA in overlay (1,0). Include the new or updated labeled common and add the new input variable to the namelist.

5) Subroutine DATA, in overlay (2,0), establishes the initial or nominal values of the variables to be used in the simulation. Every new variable must have a nominal value set, even if it is zero. Add the new common, or update existing common with new variables, in the subroutine DATA.

6) Add data statement in subroutine DATA to set nominal value of new variable.

7) Add or change common for new variable in routines where it is to be used. Add necessary coding to perform computations involving new variable.

Adding new tables.- The program has a generalized table accessing feature that allows new tables to be added without adding dimensional arrays, hard-wired table arguments, table types, table dimension, etc. The program input processor packs all tables input by the user, into an array in blank common. At execution time, the table interpolation routine, GENTAB, is directed to a particular table in the blank common array by a pointer, which is set at data initialization at the beginning of each phase. Thus, each table has a pointer associated with it. Each table also has a multiplier associated with it, by which the table is scaled during execution. To add a new table the user need only add the table pointer and the table multiplier. Because the pointer and multiplier can change from phase to phase,

they are included in the general data area of program. That is, they are defined between labeled common /IV/ and labeled /END/ as they are specified in Subroutine DICT and DATA for overlays (1,0) and (2,0), respectively. The table input processor expects to find all table pointers and multipliers together, and in pairs. The pairs must be defined between labeled common /MOTBL/ and labeled common /MOTEND/, as declared in Subroutine DICT and Subroutine DATA. A new table pointer and multiplier should be added to labeled common /MOTBL/. A new labeled common could be declared between common /MOTBL/ and /MOTEND/, but this is generally not necessary.

To add a new table the following procedure is to be followed:

1) Add two locations to labeled common /MOTBL/ for the table pointer and multiplier.

2) Replace labeled common /MOTBL/ in Subroutine DICT in overlay (1,0).

3) Add data statement in subroutine DICT to set table name in pointer, and Hollerith name of multiplier into table multiplier. For example, if a new table called EMFT is to be added, then EMFT(2) is added to common /MOTBL/. In Subroutine DICT the data statement DATA EMFT/4HEMFT, 6HEMFTM/ is added. This sets the table name, EMFT, and the table multiplier EMFTM for the input processor.

4) Table multipliers are input through namelist /TBLMLT/. Thus the table multiplier must be added to namelist /TBLMLT/ in subroutine RTBLML in overlay (1,0). Replace the labeled common /MOTBL/. Include equivalence statement to equate table multiplier with desired input name. Add input name to namelist /TBLMLT/. For example, EQUIVALENCE (EMFT(2), EMFTM); add EMFTM to namelist /TBLMLT/.

5) Replace labeled common /MOTBL/ in subroutine DATA in overlay (2,0).

6) Add data statement in subroutine DATA to set table pointer to zero and table multiplier to desired nominal value. Generally the table multiplier will be set to 1.0. For example, DATA EMFT/0 , 1.0 /.

7) To reference the new table add or replace labeled common /MOTBL/ in routine where table look-up is to be performed. To perform the interpolation, the interpolation routine GENTAB is called with the table pointer as an argument. For example:

```
VOLT = GENTAB (EMFT)
```

If the table is not input, GENTAB will return as zero.

Adding new integrals.- Any general variable computed in the simulation model can be integrated provided it satisfies the necessary conditions of differential and continuity as required by the intergration algorithms. The variables must be computed in the inner loop of the simulation, and be defined as a general variable in Subroutine DICT and Subroutine DATA. The program determines which variables are to be integrated during any phase from an integration list, which is defined in BLKDAT. The integration list contains three entries for each integral. These entries are the integral name, the derivative name, and a flag to indicate whether this integral is to be integrated or not. During phase initialization this flag can be set to turn the integration on or off. The integration list is defined in labeled common /DYNIL/ in BLKDAT. The first location in labeled common /DYNIL/ contains the total size of the list including itself. Thus, to add an integral the common /DYNIL/ must be increased by three, and the contents of DYNIL(1) increased by three.

To add a new integral the following procedure should be followed:

1) In BLKDAT, overlay (0,0) increase the dimension of DYNIL in labeled common /DYNIL/ by three for each new integral.

2) In the associated data statement increase the number prestored into DYNIL(1) by three for each integral added.

3) Add the DATA statement to set the Hollerith name of the integral, of the derivities and a nominal value of 0 or 1, depending when the integral is to be nominally off or on, into three new locations defined in DYNIL.

   For example, to add DYNPI as the integral of DYNP, the following DATA statement should appear.

```
DATA DYNIL/M, 6HTIME, 6Htime , 1
                ,      .
                ,      .
                ,      .
                  6HDYNPI, 6HDYNP , 0  /
```

4) Add integral and derivative, if required, to simulation as described previously under addition of general variables.

5) If the user desires to turn the integral on or off as a function of input, or model selected, then, the associated flag must be set in the integration list in Subroutine MOTIAL (overlay 2,2). The utility routine INTGRL can be used. The first argument is the position of the integral in the list, the second is the number of integrals to be set, and the third the flag zero or one.

Adding New Auxiliary Calculations

Additional auxiliary variable calculations can be added to the subroutine AUXFM. Their location within the routine determines when and how often these computations are performed. To add new calculations one of the following procedures should be followed:

1) Position code between the call to subroutine GAMLAM and the call to CONIC. Calculations within this area are calculated at each integration step.

2) Add variables to an existing variable partition set and add equations to the section of AUXFM which computes the variable set. The following updates must be made.

   a. Adjust the CALNAM array to reflect the additions, in common AXCAVC;

   b. Adjust the routine NOMHOL local variable array NOMVL3 to reflect the changes;

   c. Add the variable names to the NOMVL3 data statements;

   d. Adjust the NOMHOL DO loop upper limit which translate the variable names;

   e. Adjust the partition constants defined by data statements in routine DATA.

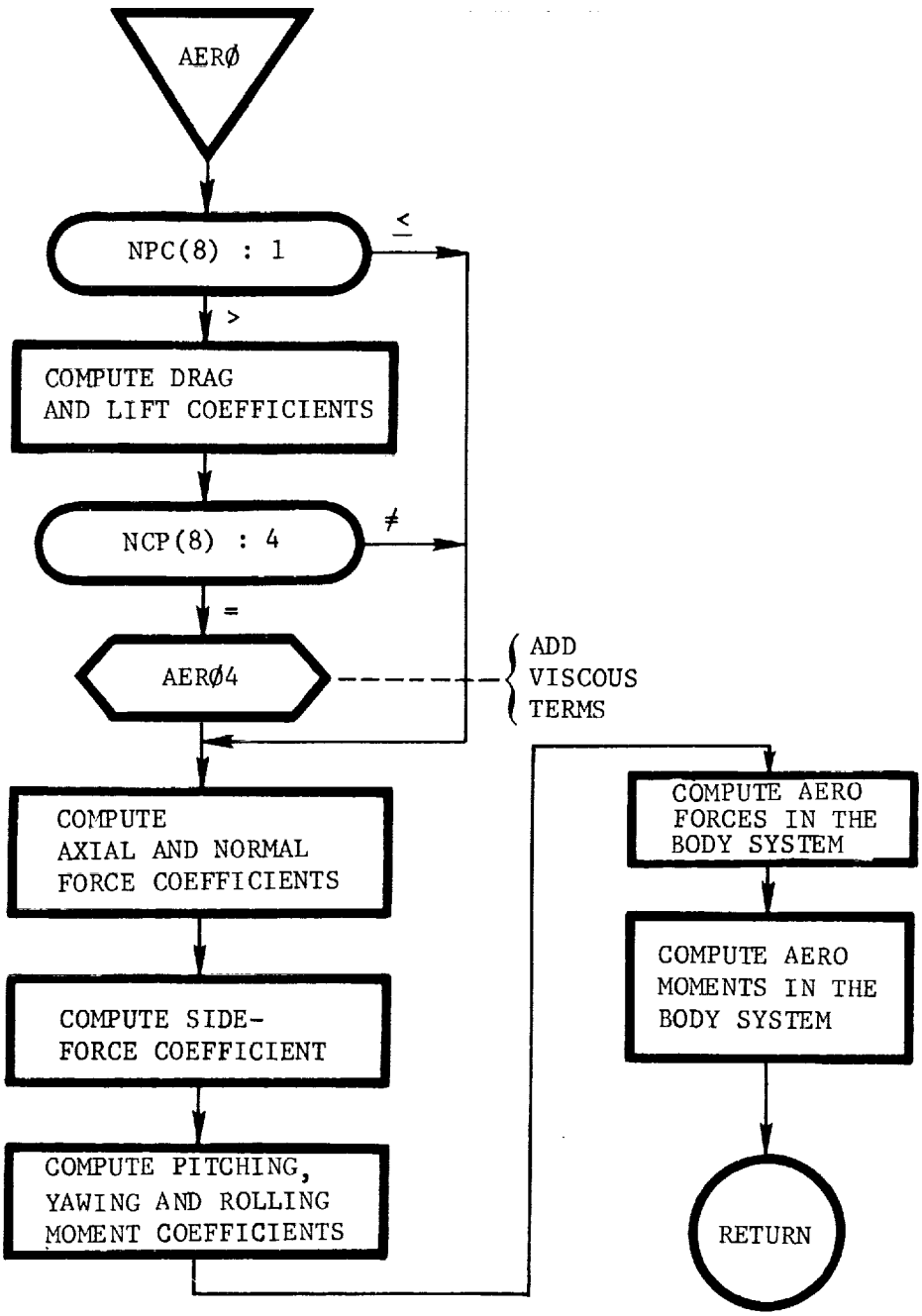
3) Define a new partition set and add equations to the end of AUXFM. Repeat steps listed in procedure 2. The ICAL and INFF variables can be used for control when these additions are computed.

III. SUBROUTINE DESCRIPTIONS, FLOW CHARTS, AND SELECTED LISTINGS

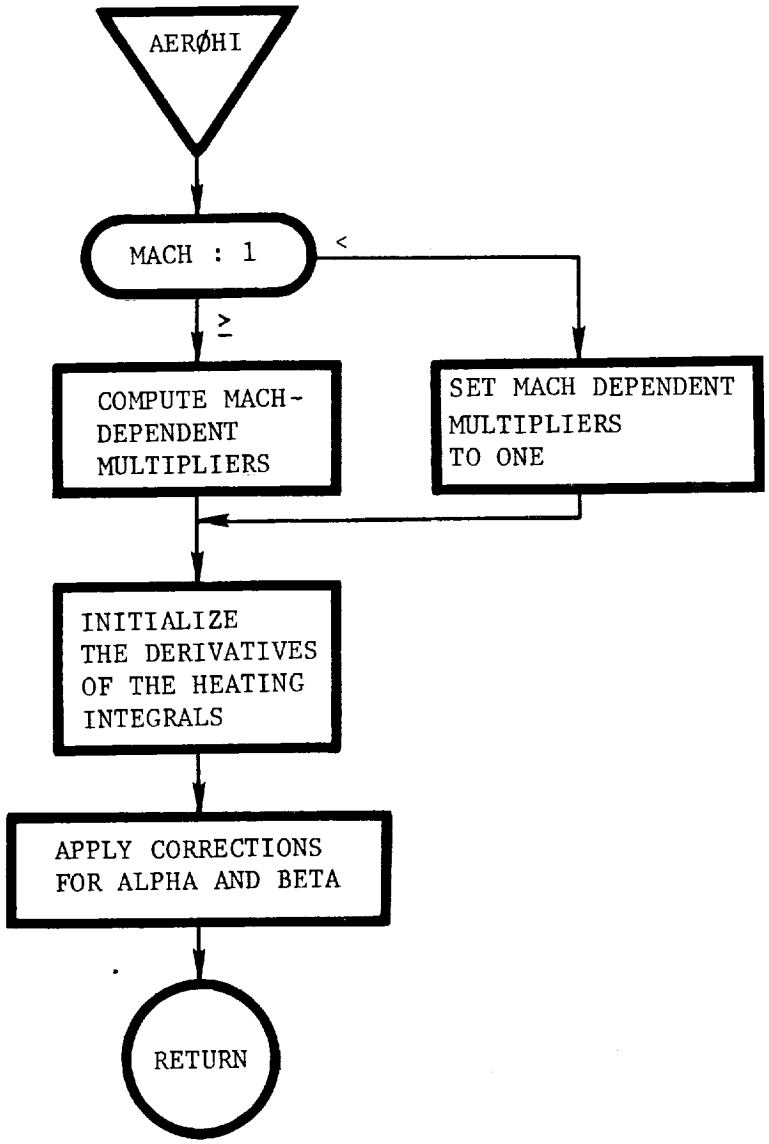
This section describes the subroutines used in the program. Flow charts and/or listings are also presented in order to show the detailed operation of the subroutines.

Note that the routines are presented alphabetically, rather than in the order shown in the previous outline of program logic. The outline enables the user to follow the program logic flow from one subroutine to another with a minimum of searching to find the next routine, but this alphabetical listing makes it easier to find subroutines at random.

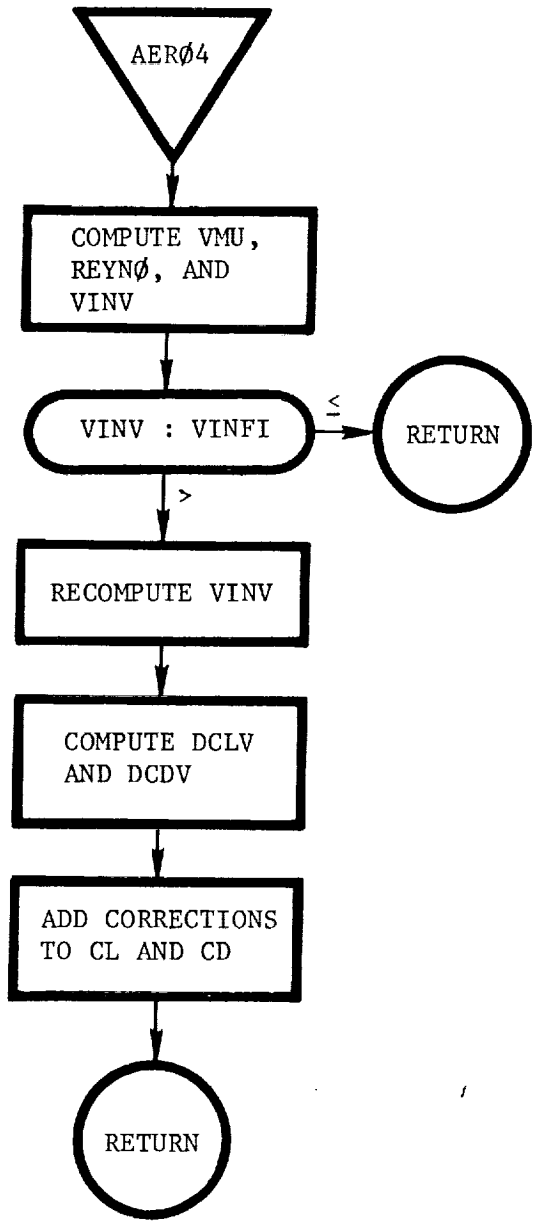
AERØ: This routine calculates the aerodynamic forces and moments in the body coordinate system.



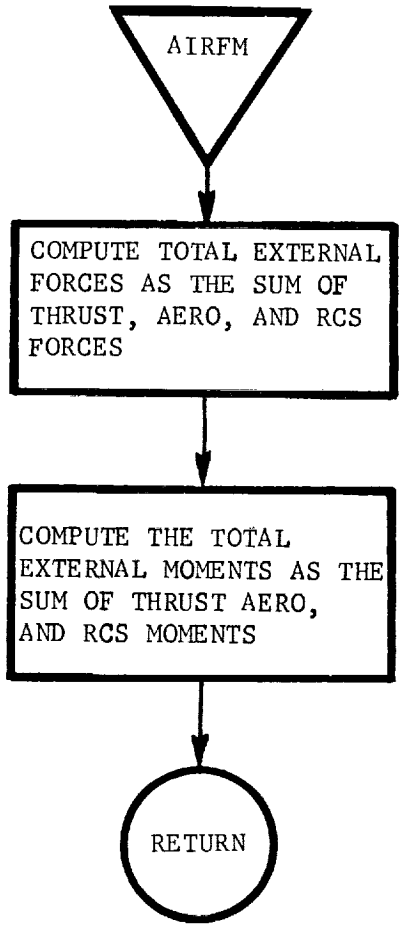
AERØHI: This routine calculates aeroheating indicators that are functions of angle of attack, sideslip, and Mach number.



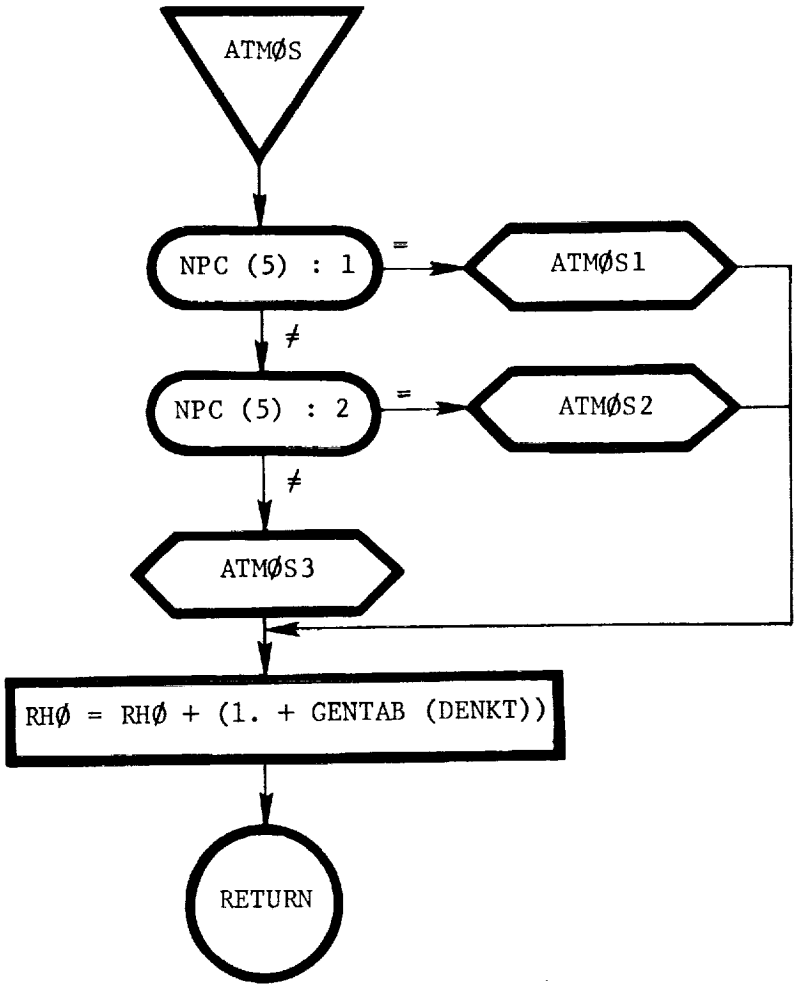
AERØ4: This routine calculates the corrections to the lift and drag coefficients (DCLV and DCDV) to be applied to CL and CD to account for viscous interaction effects.



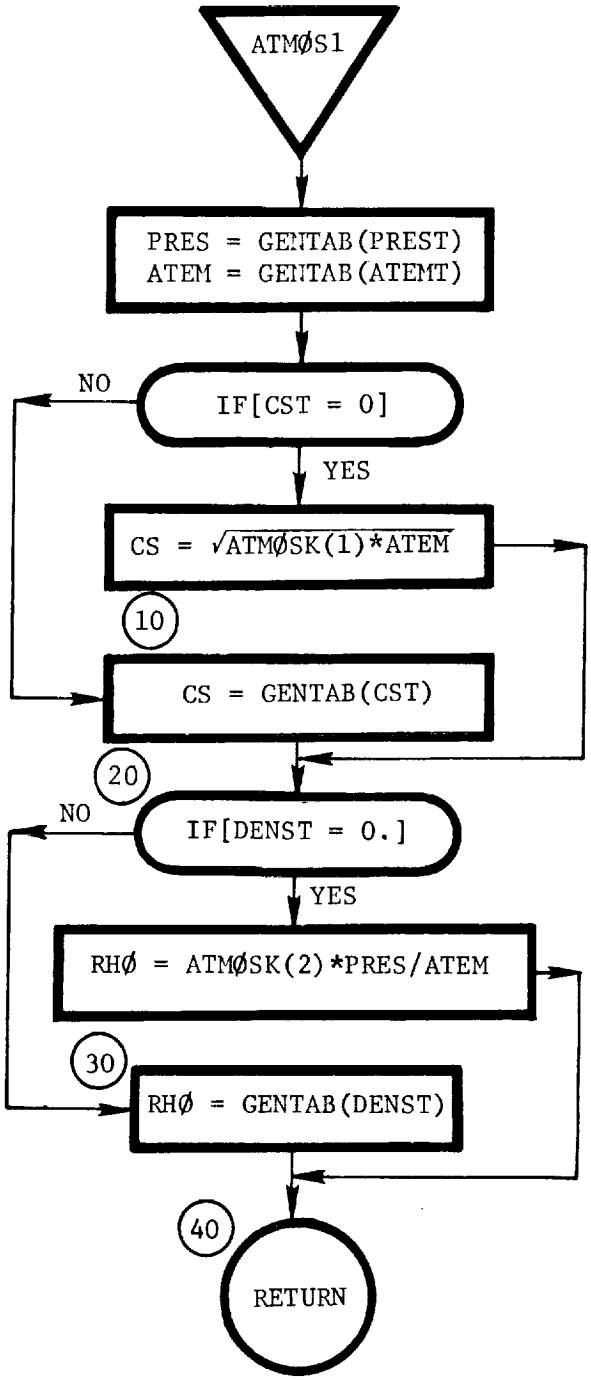
AIRFM: This routine calculates the total external forces and moments acting on the vehicle.



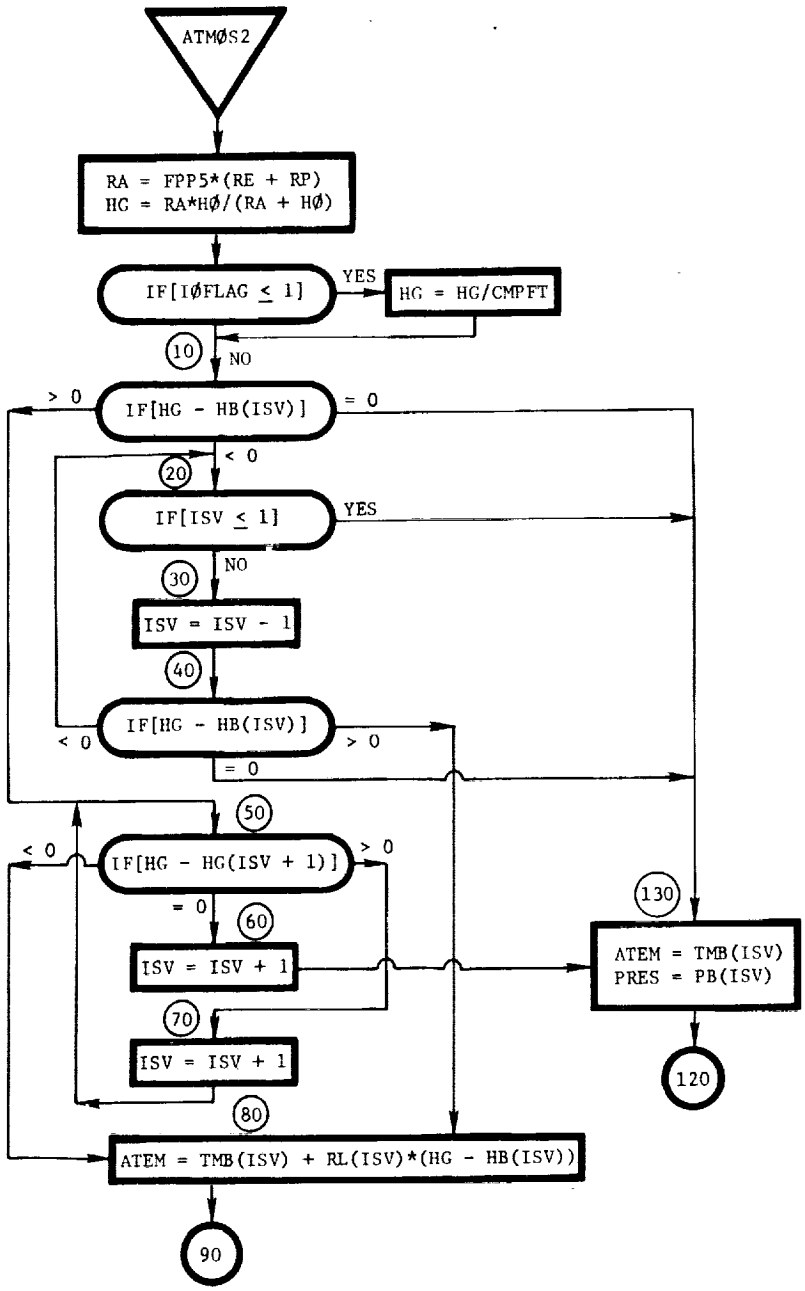
ATMØS: This routine determines which atmosphere model is to be used.

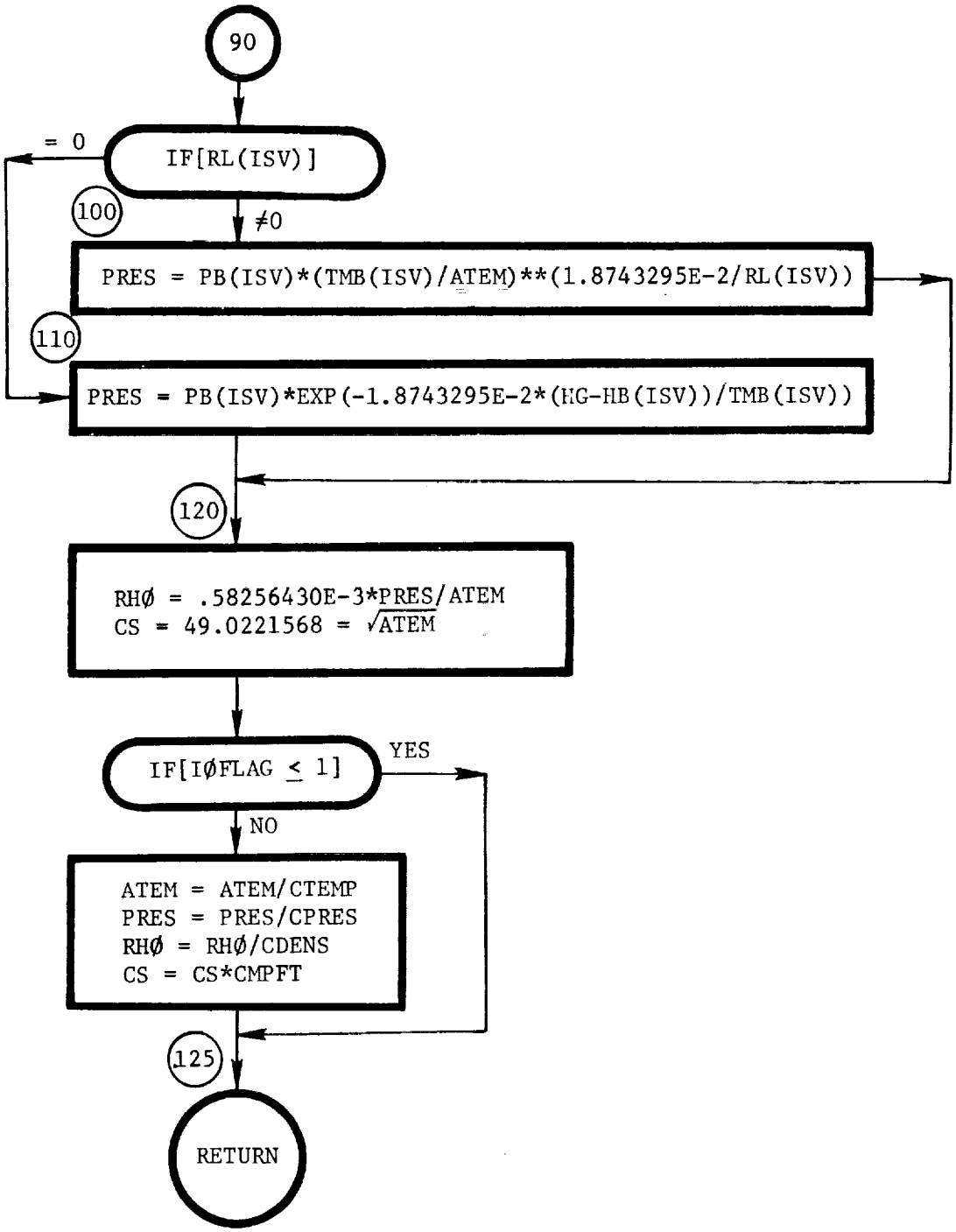


ATMØS1: This routine computes the atmospheric parameters using generalized table lookups.

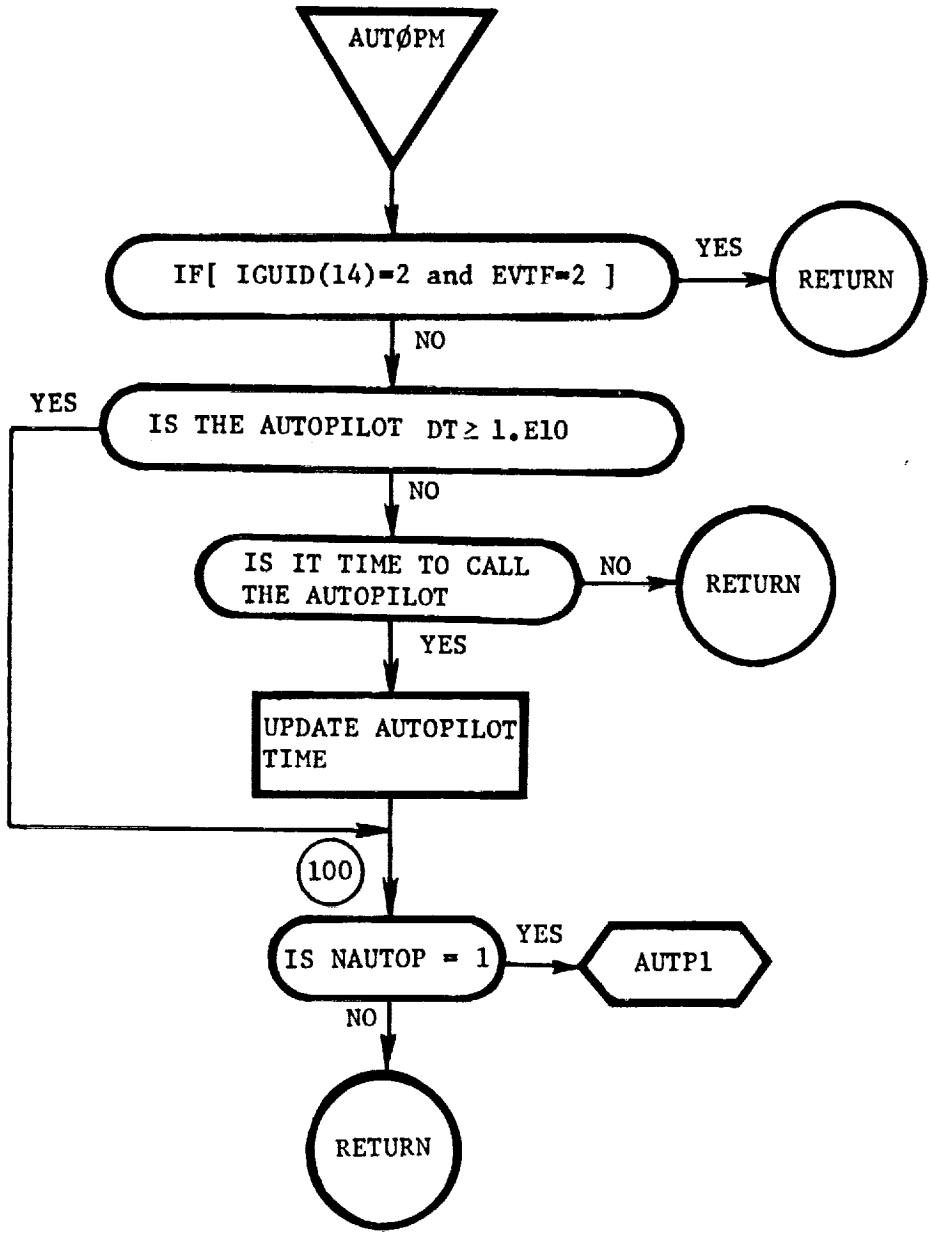


ATMØS2: This routine computes the atmospheric parameters based on the 1962 U.S. Standard atmosphere model as a function of geopotential altitude.

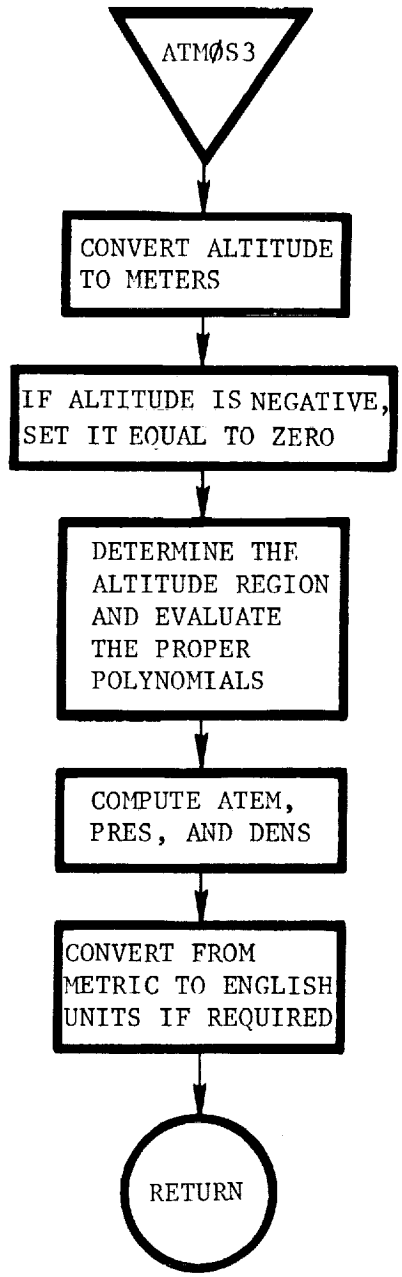



90
IF[RL(ISV)]
= 0
≠0
100
PRES = PB(ISV)*(TMB(ISV)/ATEM)**(1.8743295E-2/RL(ISV))
110
PRES = PB(ISV)*EXP(-1.8743295E-2*(HG-HB(ISV))/TMB(ISV))
120
RHØ = .58256430E-3*PRES/ATEM
CS = 49.0221568 = √ATEM
IF[IØFLAG ≤ 1]
YES
NO
ATEM = ATEM/CTEMP
PRES = PRES/CPRES
RHØ = RHØ/CDENS
CS = CS*CMPFT
125
RETURN

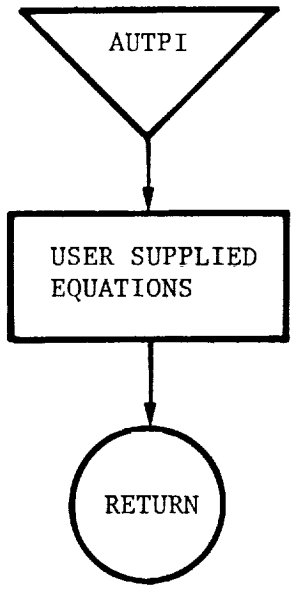
AUTØPM: This routine calls the selected autopilot model.



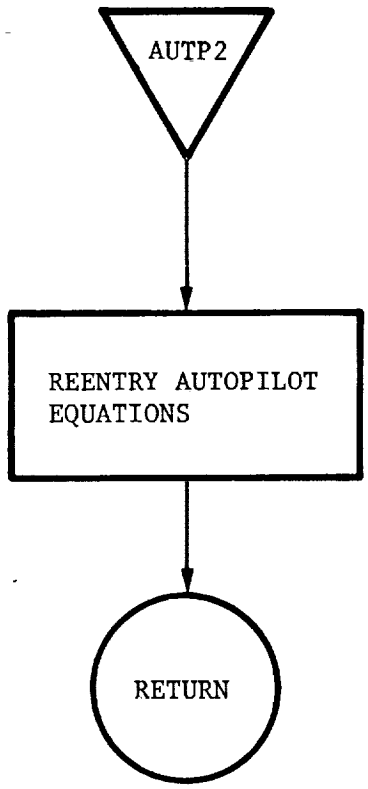
ATMØS3: This routine computes the atmospheric parameters based on the 1963 Patrick AFB atmosphere model.



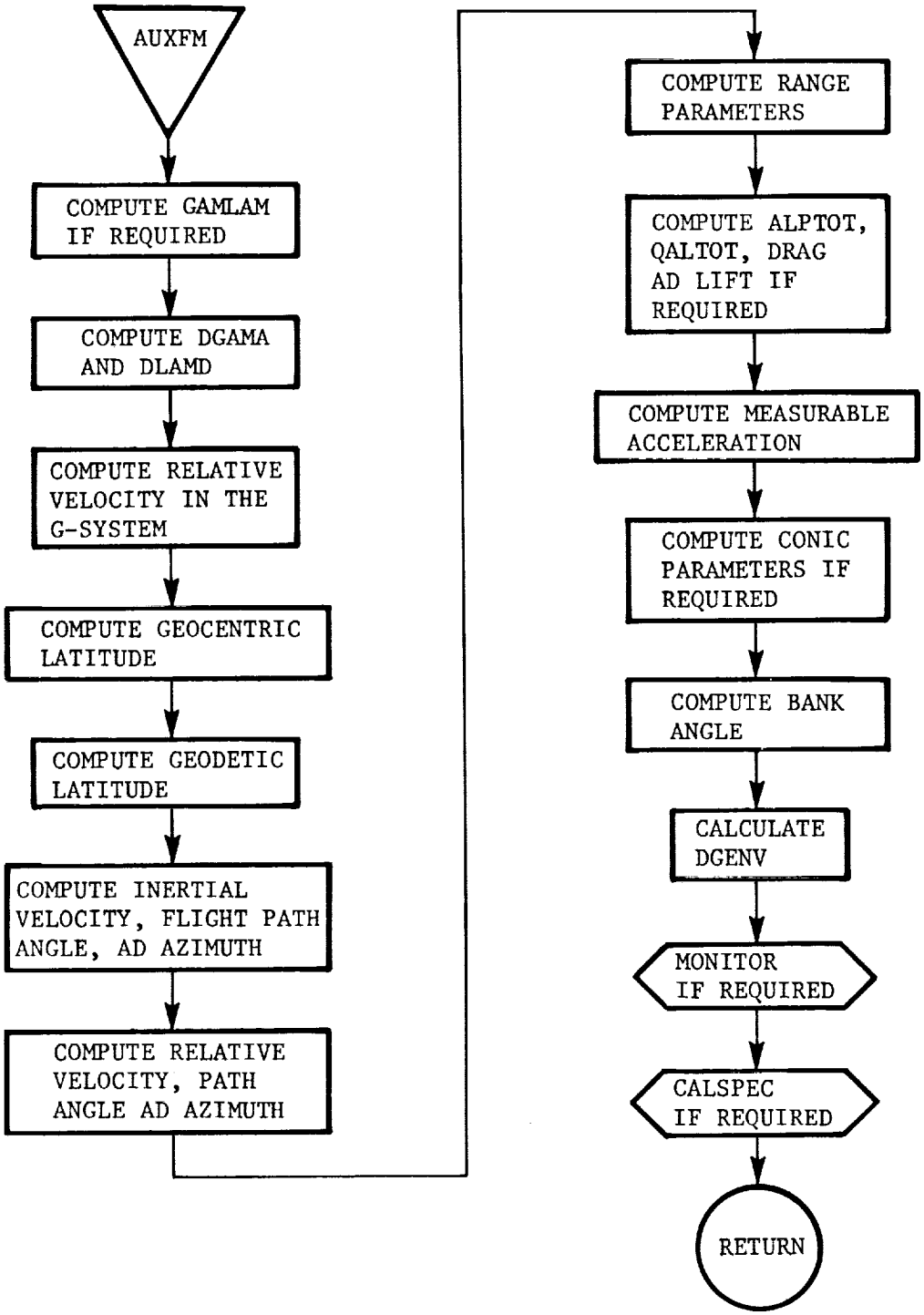
AUTP1: This is a user supplied ascent autopilot (analog with load relief) model.



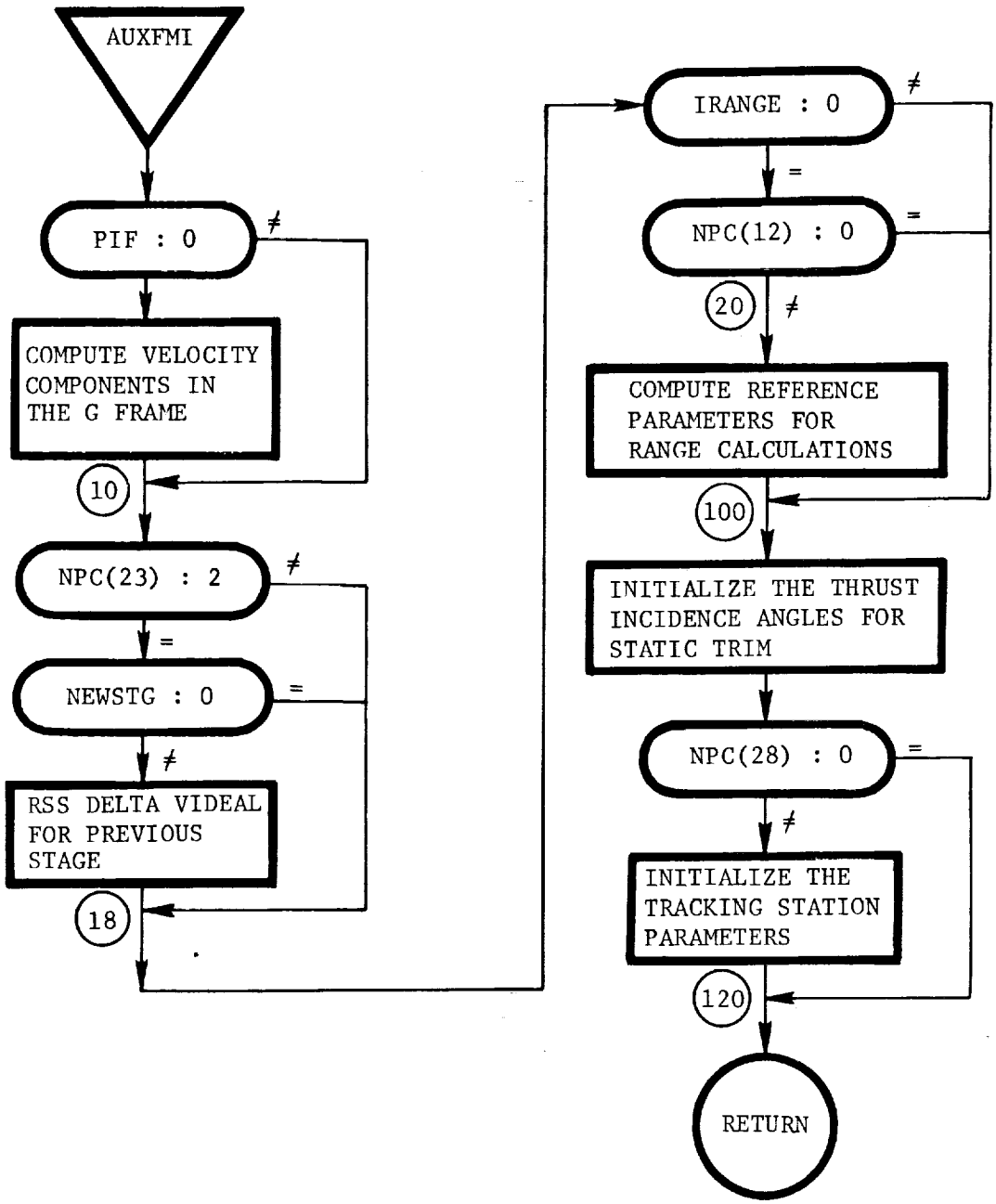
AUTP2: This is the user supplied reentry autopilot routine.



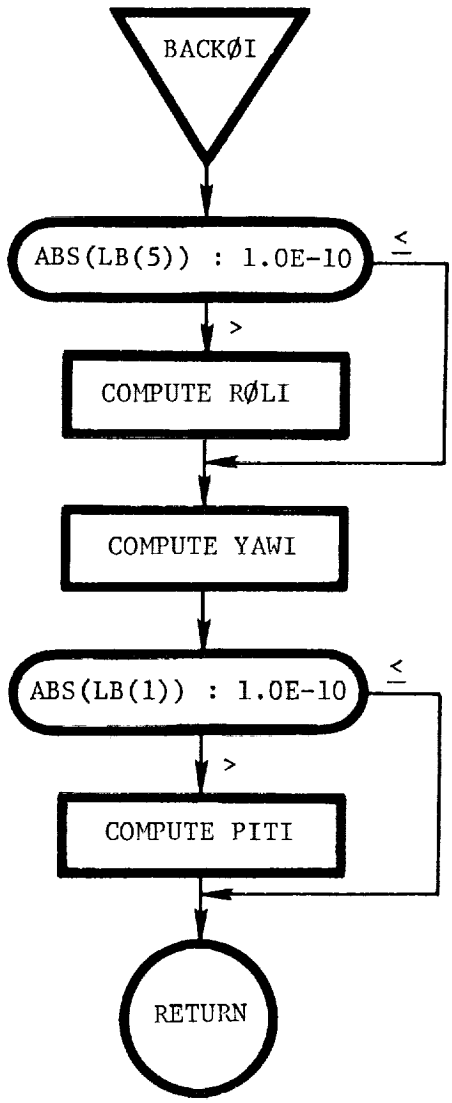
AUXFM: This routine calculates the auxiliary variables associated with the closed loop guidance equations at each integration step and all other auxiliary variables that are required as a function of input data when needed.



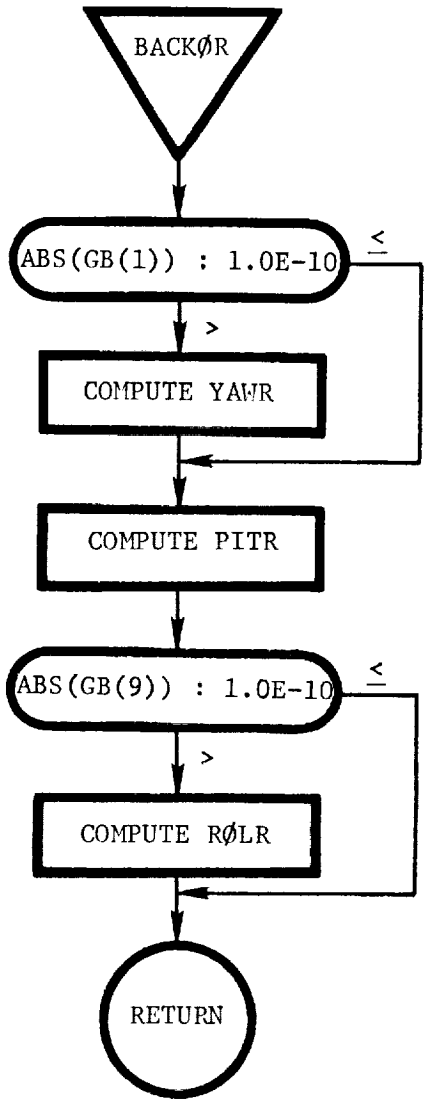
AUXFMI: This routine initializes the auxiliary calculations.



BACKØI: This routine computes the inertial Euler angles, given the LB matrix.



BACKØR: This routine computes the relative Euler angles, given the GB matrix.



BLKDAT: This routine is not called explicity but performs its function of presetting all internal program values at load time. Certain of these values can be overridden later by input if desired.

```
*DECK,BLKDAT
      BLOCK DATA
C*** BLKDAT
C              SET INITIAL PROGRAM VALUES
C*** THE DATA STATEMENTS IN THIS ROUTINE ARE STANDARDIZED
C
C
C     STANDARD FORTRAN DECLARATIONS
C              COMMON
C              DIMENSION
C              EQUIVALENCE
C              TYPE (CAN BE ANYWHERE ABOVE)
C              DATA (CAN BE ANYWHERE)
C              NAMELIST
C
C                          - W A R N I N G -
C               *** COMMON MULTRC MUST BE FIRST COMMON IN BLKDAT ***
C
*CALL MULTRC
C
*CALL PAGERC
C
*CALL SERVC
C
*CALL PADC
C
*CALL GENIC
*CALL SEARC
      COMMON/INFIV/ INFIV(1)
      COMMON / INFIC / PE(480), NPRNT
      COMMON/INFND/ INFND
      COMMON /DYNIL/ DYNIL(175)
      COMMON/INPVC/ INPCF
      COMMON IBKT(15000)
C
C
      DIMENSION DYNIL1(52)
      DIMENSION DYNIL2(48)
      DIMENSION DYNIL3(18)
      DIMENSION DYNIL4(57)
      DIMENSION PE1(216)
      DIMENSION PE2(264)
      EQUIVALENCE (DYNIL1(1),DYNIL(  1))
      EQUIVALENCE (DYNIL2(1),DYNIL( 53))
      EQUIVALENCE (DYNIL3(1),DYNIL(101))
      EQUIVALENCE (DYNIL4(1),DYNIL(119))
      EQUIVALENCE (PE(1),PE1(1))
      EQUIVALENCE (PE(217),PE2(1))
C     COMMON MULTRC
C
      DATA IN    /3     /
```

```
      DATA IO     /3      /
      DATA MULTRF/1       /
      DATA IGSIZ /0         /
C
C     COMMON PAGERC
C
      DATA ICASE /0       /
      DATA HEADER /10*1H /
C
C     COMMON SERVC
C
      DATA TEMP  /50*0            /
      DATA STEMP /25*0.0          /
      DATA IR1   /0               /
      DATA IR2   /0               /
      DATA IR3   /0               /
      DATA IR4   /0               /
      DATA IR5   /0               /
      DATA IR6   /0               /
      DATA NULL  /1HU             /
      DATA N00   /0               /
      DATA N01   /1               /
      DATA N02   /2               /
      DATA N03   /3               /
      DATA N04   /4               /
      DATA N05   /5               /
      DATA N06   /6               /
      DATA N07   /7               /
      DATA N08   /8               /
      DATA N09   /9               /
      DATA N10   /10              /
      DATA N11   /11              /
      DATA N12   /12              /
      DATA N13   /13              /
      DATA N14   /14              /
      DATA N15   /15              /
      DATA FP00  /0.0             /
      DATA FPP5  / .5             /
      DATA FP1   /1.0             /
      DATA FP2   /2.0             /
      DATA FP3   /3.0             /
      DATA FP4.  /4.0             /
      DATA FP5   /5.0             /
      DATA FP6   /6.0             /
      DATA FP7   /7.0             /
      DATA FP8   /8.0             /
      DATA FP9   /9.0             /
      DATA FP10  /10.0            /
      DATA FP11  /11.0            /
      DATA FP12  /12.0            /
      DATA FP13  /13.0            /
```

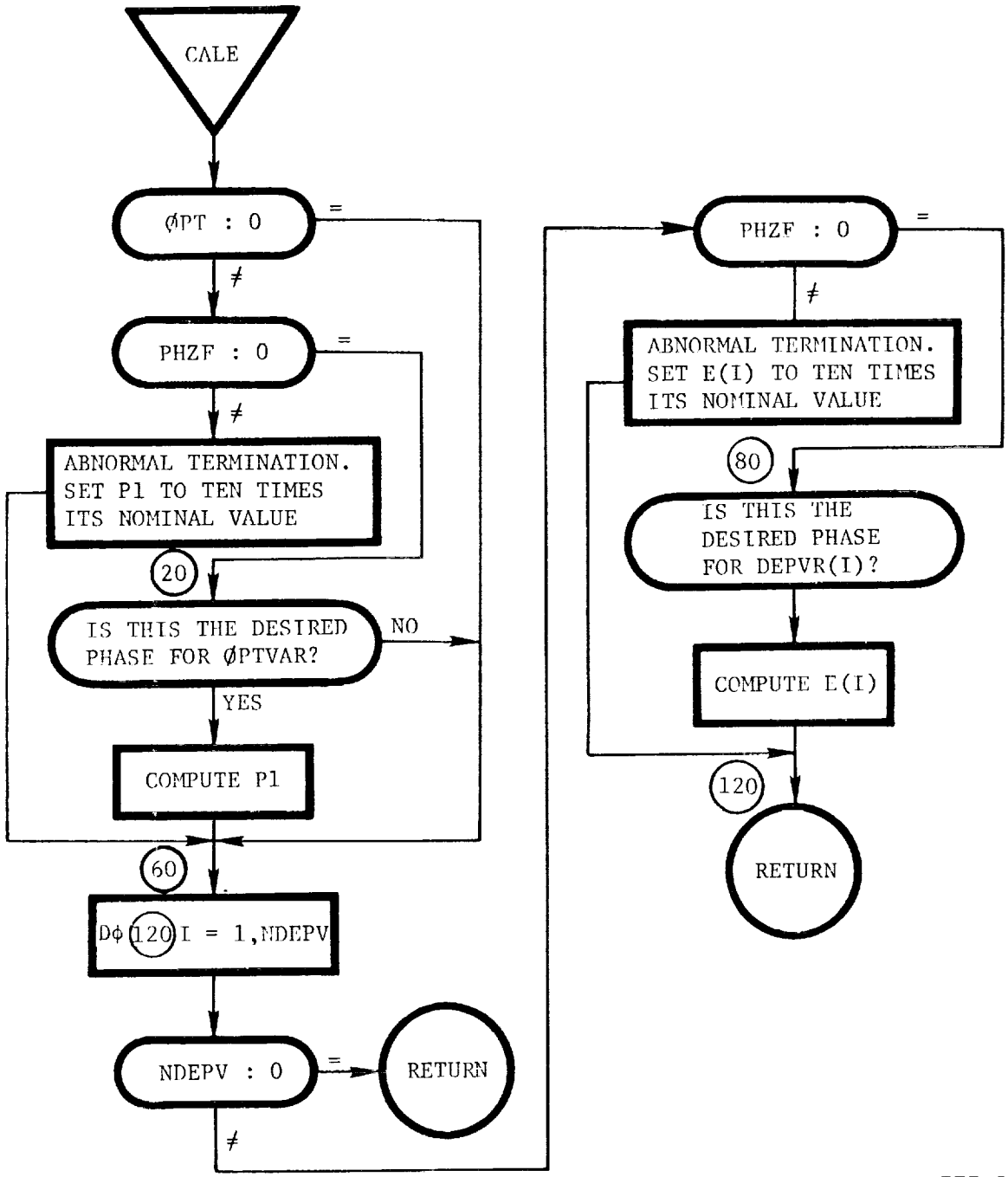
```
      DATA FP14  /14.0            /
      DATA FP15  /15.0            /
      DATA FP60  /60.0            /
      DATA FP90  /90.0            /
      DATA FP180 /180.0           /
      DATA FP270 /270.0           /
      DATA FP360 /360.0           /
      DATA PI02  /1.5707963267948965/
      DATA PI    /3.1415926535897932/
      DATA RPD   /0.01745329251994329/
      DATA DPR   /57.29577951308232/
      DATA TWOPI /6.283185307179586/
      DATA FTPNM  /6076.1155      /
      DATA CMPFT  /.3048          /
      DATA IOFLAG /0/
C*** CFORCE = NEWTONS PER POUND
      DATA CFORCE /4.4482216152605/
C***  CPRES = LB/FT**2 PER NEWTONS/METERS**2
      DATA CPRES /.0208854347/
C***  CTEMP = DEGREES F PER DEGREES K
      DATA CTEMP  /1.8            /
C*** CDENS = SLUGS/FT**3 PER KILOGRAM/METER**3
      DATA CDENS /.00194031965 /
C***  CHEAT= JOULES PER BTU
      DATA CHEAT /1054.350264488888 /
C***  CMASS = KILOGRAMS PER SLUG
      DATA CMASS /14.5939029 /
      DATA CVDIST/6076.1155       /
      DATA IDENT /1.0    ,0.0    ,0.0    ,
     1            0.0    ,1.0    ,0.0    ,
     2            0.0    ,0.0    ,1.0    /
      DATA IVSZ  /0               /
      DATA XINF  /1.0E+11           /
      DATA SPDY  /86400.          /
      DATA SPHR  /3600.           /
      DATA FPEM6 /1.0E-6          /
      DATA FPEM8 /1.0E-8          /
      DATA FPEM10/1.0E-10         /
C
      DATA INFIV /0               /
      DATA INPCF /0                /
C***
C
                                 DATA PE1
     1/0,6HTIME  ,0,6HTIMES ,0,6HTDURP ,0,6HDENS  ,0,6HPRES  ,0,6HATEM
     2,0,6HALTITO,0,6HGCRAD ,0,6HGDLAT ,0,6HGCLAT ,0,6HLONG  ,0,6HLONGI
     3,0,6HVELI  ,0,6HGAMMAI,0,6HAZVELI,0,6HXI    ,0,6HVXI   ,0,6HAXI
     4,0,6HVELR  ,0,6HGAMMAR,0,6HAZVELR,0,6HYI    ,0,6HVYI   ,0,6HAYI
     5,0,6HVELA  ,0,6HGAMMAA,0,6HAZVELA,0,6HZI    ,0,6HVZI   ,0,6HAZI
     6,0,6HGAMAD ,0,6HAZVAD ,0,6HDWNRNG,0,6HCRRNG ,0,6HDPRNG1,0,6HDPRNG2
     7,0,6HTHRUST,0,6HWEIGHT,0,6HWDOT  ,0,6HWEICON,0,6HWPROP ,0,6HASMG
```

```
      8,0,6HFTXB   ,0,6HFAXB   ,0,6HAXB    ,0,6HTMXB   ,0,6HAMXB   ,0,6HTTMXB
      9,0,6HFTYB   ,0,6HFAYB   ,0,6HAYB    ,0,6HTMYB   ,0,6HAMYB   ,0,6HTTMYB
      D,0,6HFTZB   ,0,6HFAZB   ,0,6HAZB    ,0,6HTMZB   ,0,6HAMZB   ,0,6HTTMZB
      A,0,6HCA     ,0,6HCLL    ,0,6HCD     ,0,6HDRAG   ,0,6HDYNP   ,0,6HASXI
      B,0,6HCY     ,0,6HCM     ,0,6HCL     ,0,6HLIFT   ,0,6HMACH   ,0,6HASYI
      C,0,6HCN     ,0,6HCW     ,0,6HHEATRT,0,6HTLHEAT,0,6HREYNO  ,0,6HASZI
      D,0,6HROLBD  ,0,6HROLBDD,0,6HROLI   ,0,6HYAWR   ,0,6HALPHA  ,0,6HALPTOT
      E,0,6HPITBD  ,0,6HPITBDD,0,6HYAWI   ,0,6HPITR   ,0,6HBETA   ,0,6HQALPHA
      F,0,6HYAWBD  ,0,6HYAWBDD,0,6HPITI   ,0,6HROLR   ,0,6HBNKANG,0,6HALPDOT
      G,0,6HROLBER,0,6HROLAC  ,0,6HXREF   ,0,6HXCG    ,0,6HIXX    ,0,6HIXY
      H,0,6HPITBER,0,6HPITAC  ,0,6HYREF   ,0,6HYCG    ,0,6HIYY    ,0,6HIXZ
      I/
                                DATA PE2
      1/0,6HYAWBER,0,6HYAWAC  ,0,6HZREF   ,0,6HZCG    ,0,6HIZZ    ,0,6HIYZ
      2 ,252*0
      3/
       DATA NPRNT /240/
C
C
C
       DATA DYNIL1
      0 /175
      1 ,6HTIME   ,6HDTIME ,1
      2 ,6HXI     ,6HVXI   ,1
      3 ,6HYI     ,6HVYI   ,1
      4 ,6HZI     ,6HVZI   ,1
      5 ,6HVXI    ,6HAXI   ,1
      6 ,6HVYI    ,6HAYI   ,1
      7 ,6HVZI    ,6HAZI   ,1
      8 ,6HMASS   ,6HDMASS ,1
      9 ,6HE0     ,6HDE0   ,1
      0 ,6HE1     ,6HDE1   ,1
      A ,6HE2     ,6HDE2   ,1
      B ,6HE3     ,6HDE3   ,1
      C ,6HFVAL1  ,6HDFVAL1,0
      D ,6HFVAL2  ,6HDFVAL2,0
      E ,6HFVAL3  ,6HDFVAL3,0
      F ,6HTLHEAT,6HHEATRT,1
      G ,6HTIME   ,6HDTIME ,0
      H /
       DATA DYNIL2
      1/6HPWPROP,6HPWDOT  ,0
      2,6HHTURB ,6HHTURBD,0
      3,6HTIMRF1,6HDTIMR1,0
      4,6HTIMRF2,6HDTIMR2,0
      5,6HTIMRF3,6HDTIMR3,0
      6,6HTIMRF4,6HDTIMR4,0
      7,6HGINT1 ,6HGDERV1,0
      8,6HGINT2 ,6HGDERV2,0
      9,6HGINT3 ,6HGDERV3,0
      0,6HGINT4 ,6HGDERV4,0
```

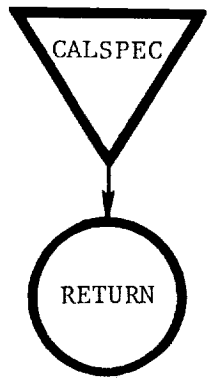
```
      A,6HGINT5 ,6HGDERV5,0
      B,6HGINT6 ,6HGDERV6,0
      C,6HGINT7 ,6HGDERV7,0
      D,6HGINT8 ,6HGDERV8,0
      E,6HGINT9 ,6HGDERV9,0
      F,6HGINT10,6HGDERV0,0
      G /
       DATA DYNIL3
      1 /6HROLB  ,6HROLBD ,1
      2 ,6HPITB  ,6HPITBD ,1
      3 ,6HYAWB  ,6HYAWBD ,1
      4 ,6HROLBD ,6HROLBDD,1
      5 ,6HPITBD ,6HPITBDD,1
      6 ,6HYAWBD ,6HYAWBDD,1
      7 /
       DATA DYNIL4
      1 /6HTROLN ,6HTROLND,0
      2 ,6HTPITN ,6HTPITND,0
      3 ,6HTYAWN ,6HTYAWND,0
      4 ,6HTROLP ,6HTROLPD,0
      5 ,6HTPITP ,6HTPITPD,0
      6 ,6HTYAWP ,6HTYAWPD,0
      7 ,6HALPERI,6HALPERR,0
      8 ,6HWPCONJ,6HWDOTJ ,0
      9 ,6HDSPINT,6HDSPD  ,0
      0 ,6HDSYINT,6HDSYD  ,0
      A ,6HROLBC ,6HROLBDC,0
      B ,6HPITBC ,6HPITBDC,0
      C ,6HYAWBC ,6HYAWBDC,0
      D ,6HEA2I  ,6HEA2   ,0
      E ,6HEY5I  ,6HEY5   ,0
      F ,6HTONT  ,6HTONTD ,0
      G ,6HDELA  ,6HDELAD ,0
      H ,6HDELE  ,6HDELED ,0
      I ,6HDELR  ,6HDELRD ,0/
C
C      COMMON SEARC
C
       DATA CONEPS/      89.9    ,
      1                  3*.00001,
      2                  2*.00001 /
       DATA CONSEX/  .000001
      1            ,     .001    /
       DATA DEPPH / 25*900.0    /
       DATA DEPTL /   25*1.0    /
       DATA DEPVAL/   25*0.0    /
       DATA DEPVR /   25*0.0    /
       DATA FITERR/  .000001
      1            ,     .001    /
       DATA GAMAX   /10.0    /
       DATA IDEB  /         0    /
```

```
      DATA IDEPVR/      25*0    /
      DATA IFDEG /      25*0    /
      DATA INDPH /      25*0    /
      DATA INDVR /      25*0    /
      DATA IPRO  /         0    /
      DATA LIMIT /5000B       /
      DATA MAXITR/        10    /
      DATA MODEW /         1    /
      DATA NDEPV /         0    /
      DATA NINDV /         0    /
      DATA OPT   /         0    /
      DATA OPTPH /900.          /
      DATA OPTVAR/         0    /
      DATA PCTCC /.1             /
      DATA PCTOLD/.1             /
      DATA PERT  /25*1.E-4        /
      DATA PGEPS /      1.0    /
      DATA P2MIN /       1.0    /
      DATA SRCHM /         0    /
      DATA STMINP/        .1
     1           ,        .1    /
      DATA STPMAX/  1.0E+10    /
      DATA U     /   25*0.0    /
      DATA WCON  /    100.0    /
      DATA WOPT  /       1.0    /
      DATA ISFLG /63              /
      DATA IWTFLG/0               /
      DATA WVU   /25*1.0           /
      DATA ITOPF /0                /
      DATA STPMXS/1HU/
      DATA ISENS /0/
      DATA STPMXO/1.0E10          /
C
C     COMMON GENIC
C
      DATA IPRT  /63              /
      DATA LISTIN/2               /
      DATA ESNI /1.0/
C
C     COMMON PADC
C
      DATA NPAD  /9.,4.,14.44943979/
      DATA PDLMAX /2.0/
      DATA SIGDEL/0.0             /
      DATA SDIF  /0.0             /
      END
```

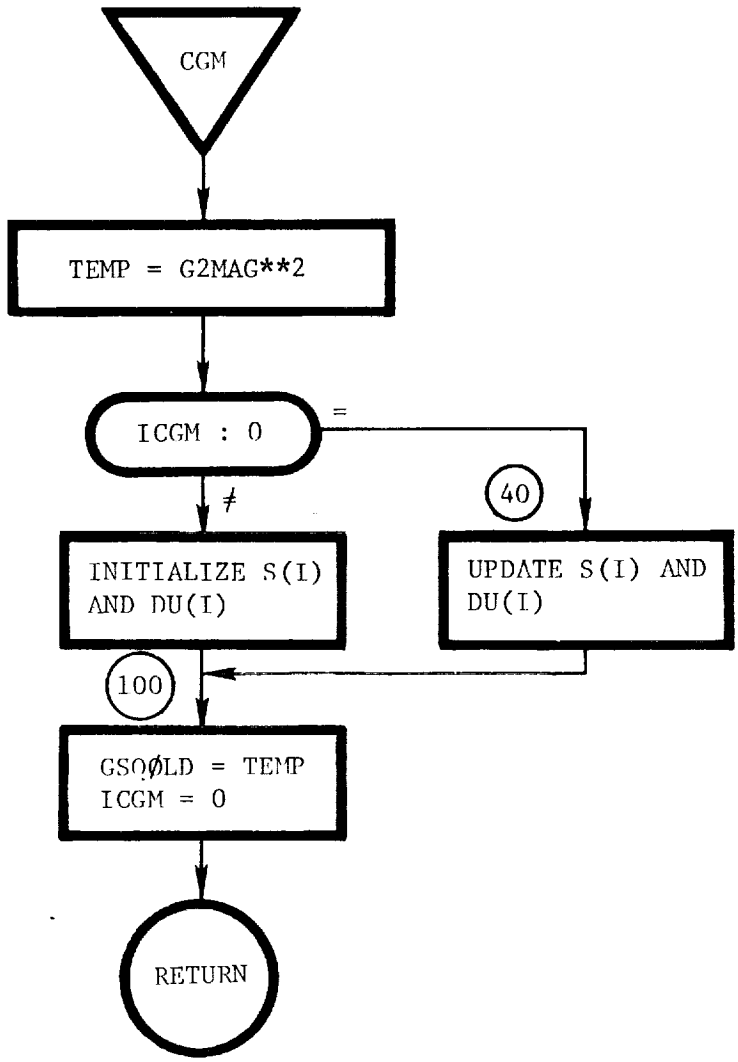
CALE: This routine calculates the performance index (P1) and the error in the target variables (E(I)).



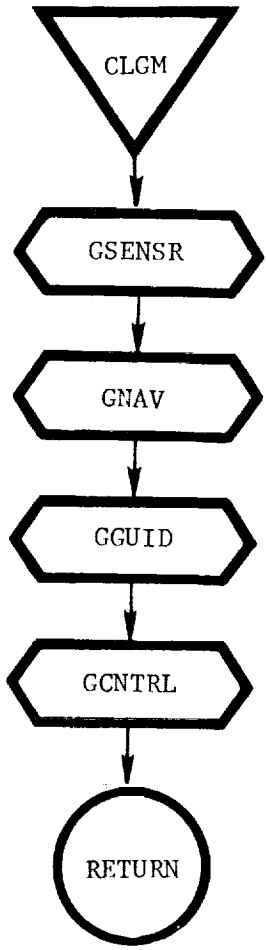
CALSPEC: This routine is a blank routine to be used when special calculations of a temporary nature are required. This routine is called at the end of each integration step from AUXFM.



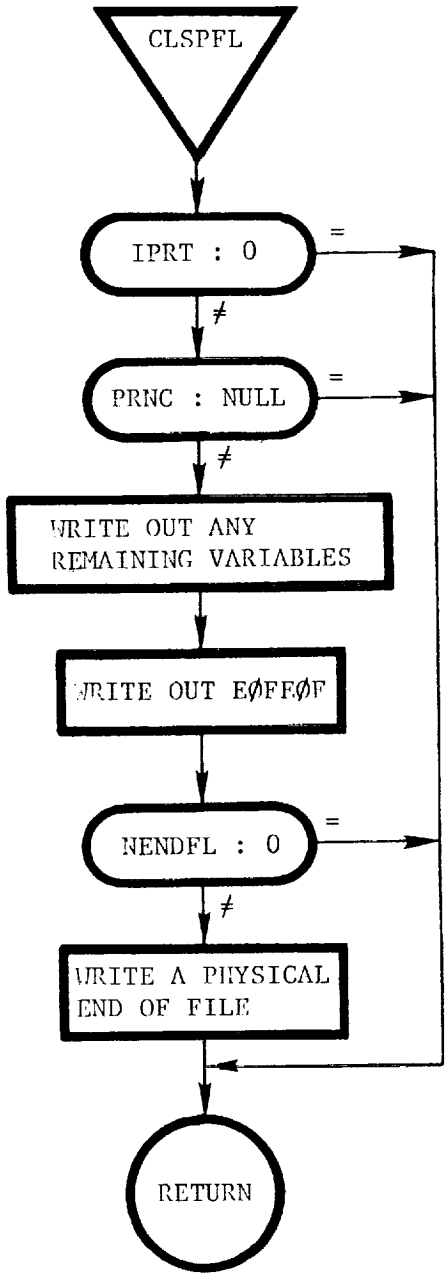
CGM: This routine contains the logic for the conjugate gradient method. It is a second-generation unconstrained optimization technique that has the stability of the steepest-descent method and the convergence properties of the second-order techniques.



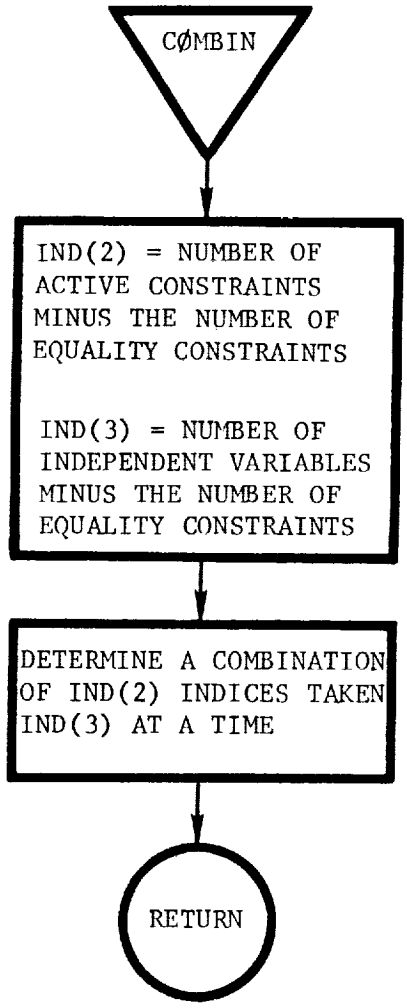
CLGM: This routine contains the executive logic for the closed-loop guidance routines.



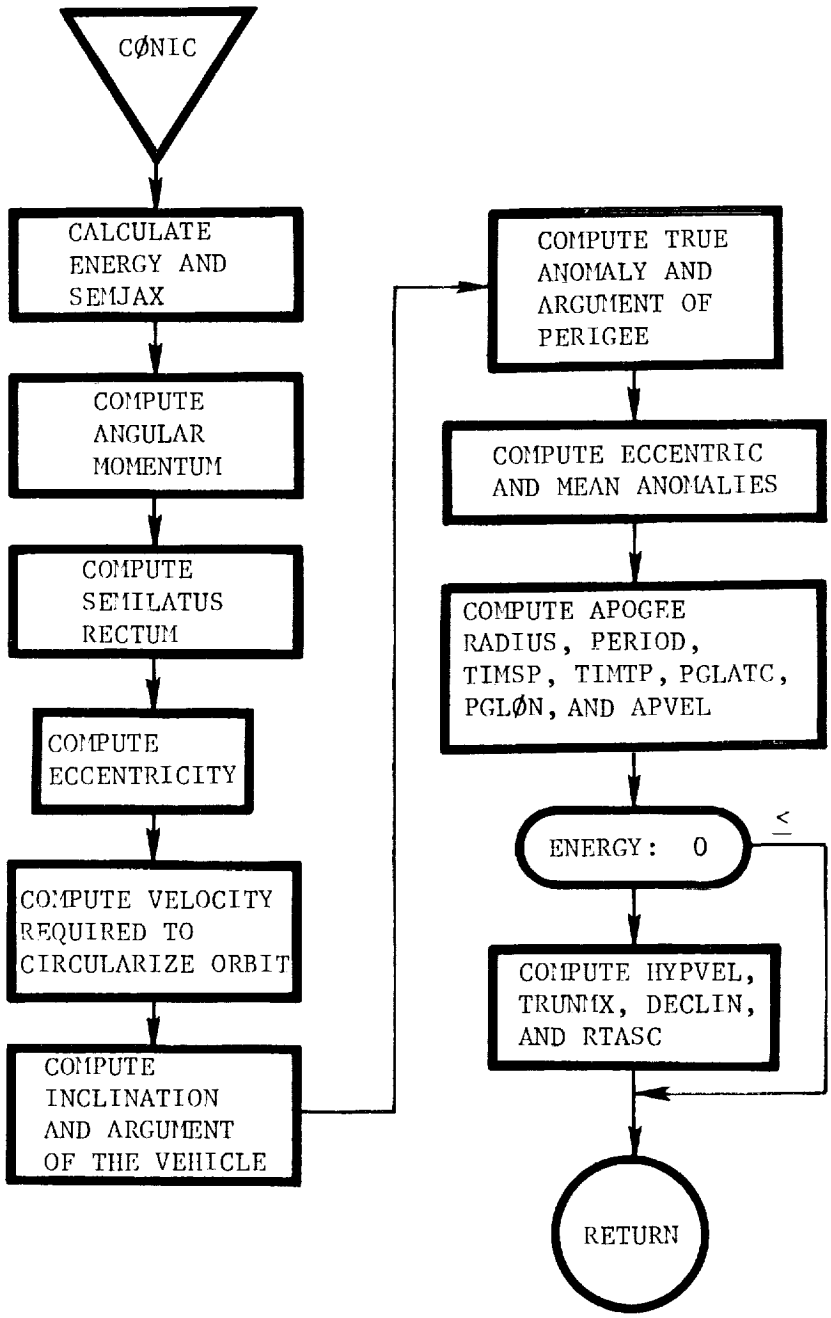
CLSPFL (NENDFL): This routine closes out profile records and writes a physical end of file on the profile tape for each trajectory.



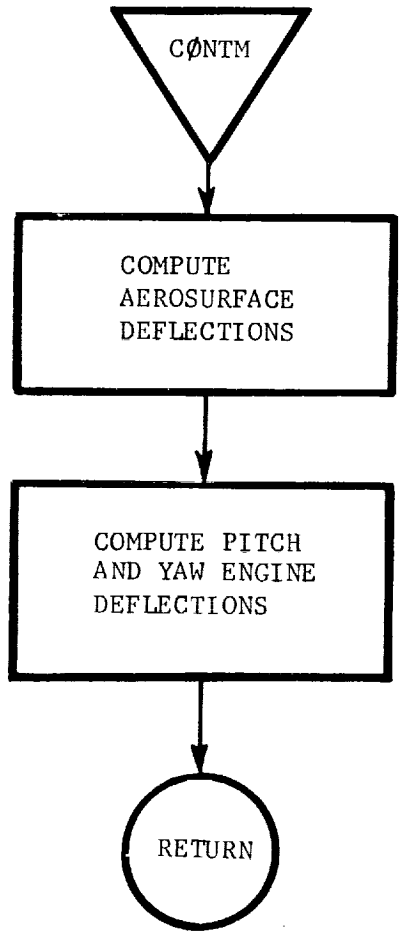
CØMBIN: This routine determines all combinations of the indices of the active constraints. These combinations of constraints are used to determine if any constraints can be dropped, and are used when the number of tight constraints exceeds the number of independent variables.



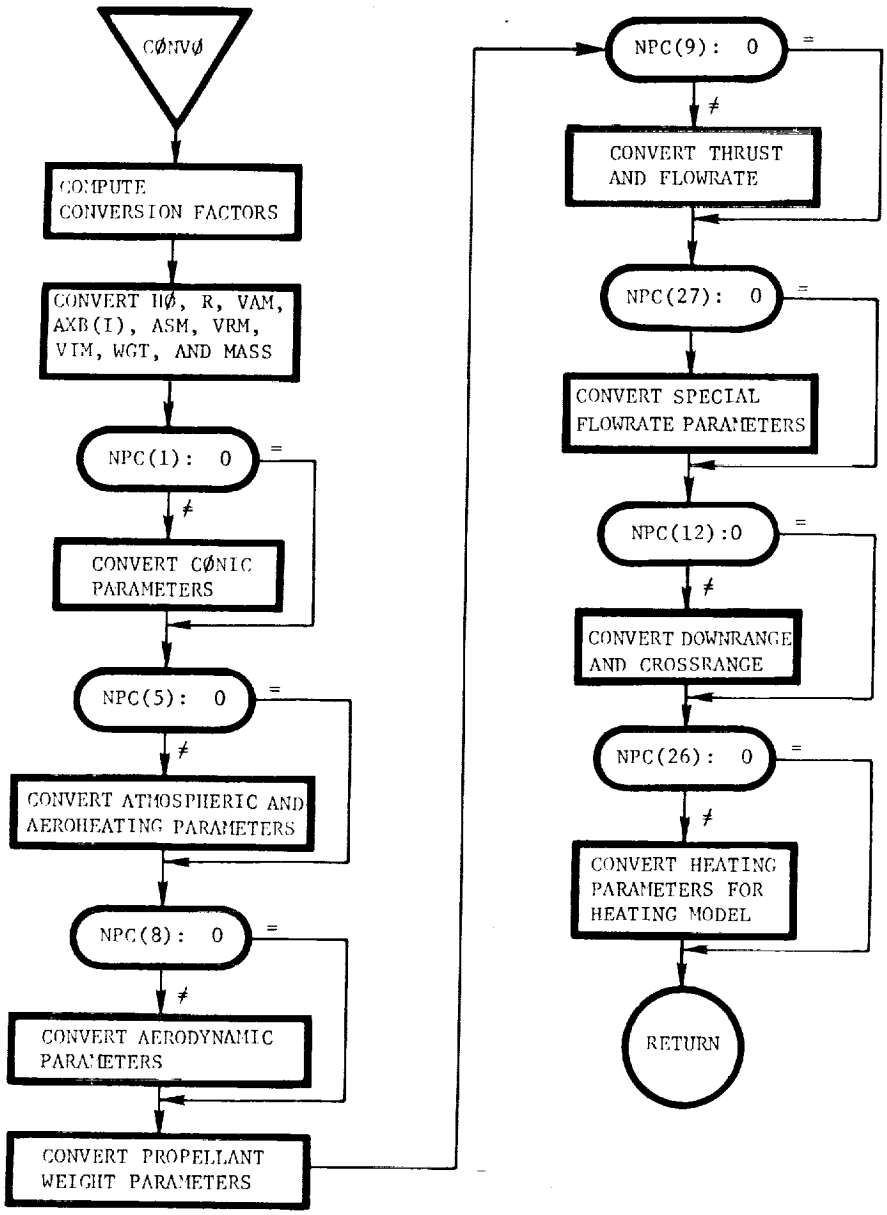
CØNIC: This routine calculates the Keplerian conic for either elliptic or hyperbolic orbits, based on the value of the orbital energy.



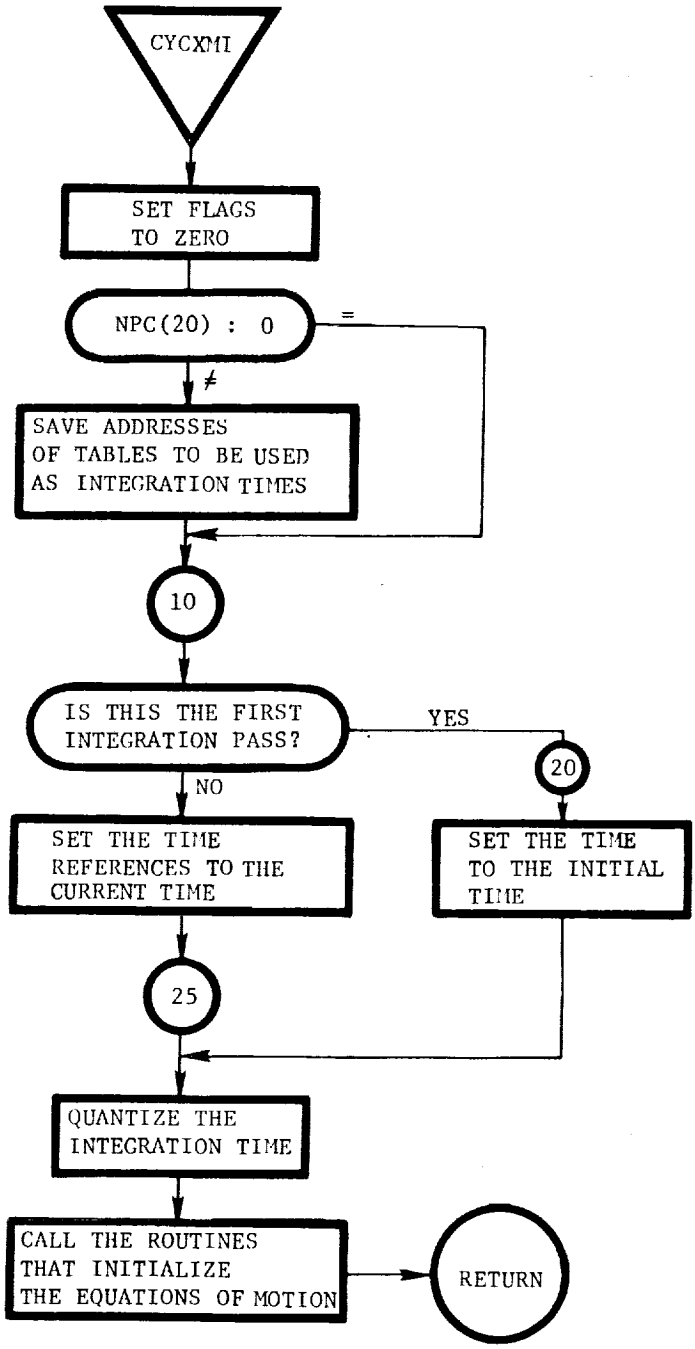
CØNTM: This routine calculates engine and aerodynamic control surface deflections based upon the autopilot commands and the mixing logic.



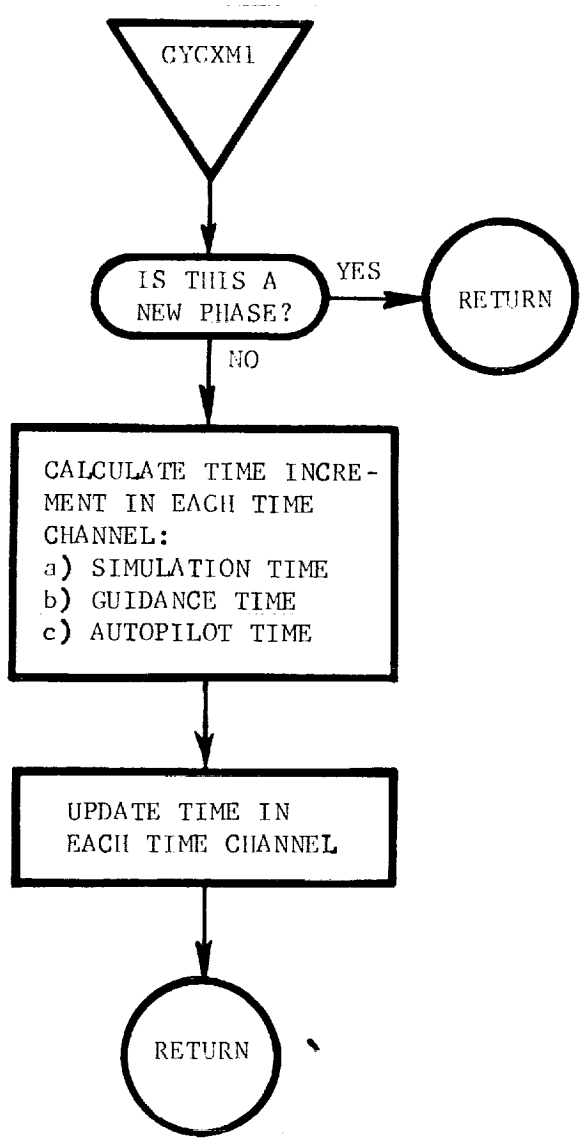
CØNVØ: This routine converts the output variables from metric to English, or vice-versa, as required.



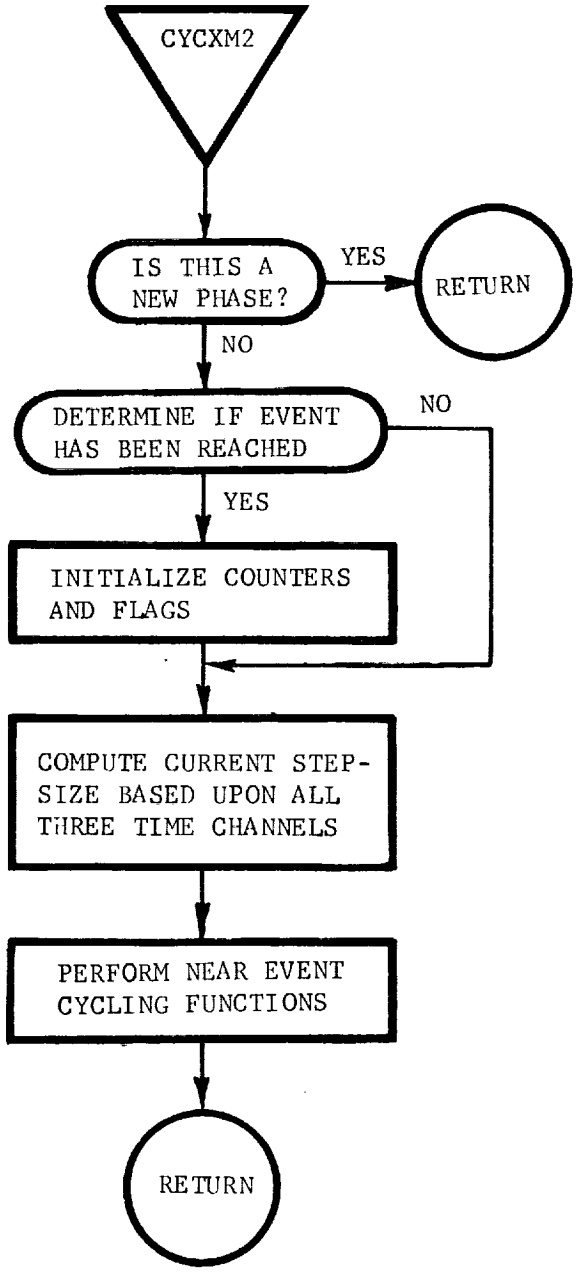
CYCXMI: This routine initializes the program at the beginning of each new cycle.



CYCXM1: This routine updates the three time channels.



CYCXM2: This routine computes the next integration step-size base upon all three time channels.



DATA: This routine is not called explicitly, but presets the program variables to their stored values at the time overlay (2,0) is called.

```
*DECK,DATA
      SUBROUTINE DATA
C*** DATA
C                 DATA - DEFINES COMPUTATIONAL COMMONS
C                        (IV - END) INITIAL DATA VALUES
C*** THE DATA STATEMENTS IN THIS ROUTINE ARE STANDARDIZED
C
C
      COMMON/IV/ IV(2)
C
*CALL AUTOPC
*CALL AUXVC
*CALL CONICC
*CALL CYCVC
*CALL CCONA1
*CALL DPGVC
*CALL DYNVC
*CALL INFVC
*CALL MNMMLT
*CALL HOLINC
*CALL MOTBL
*CALL MOTIC
*CALL RMOTIC
*CALL MOTVC
*CALL RMOTVC
*CALL PHZVC
*CALL RANGEC
*CALL SAUTOC
*CALL SPECAL
*CALL TGOVC
*CALL GUIDIC
*CALL GUIDVC
C
      COMMON/END/ END
C
C
C
*CALL AXCALC
C
*CALL DYTEM
*CALL LOCAL
                                    COMMON/PRO/
     1  PRO(502)
C
      COMMON /GFLAGS/ IGFLAG(10)
      COMMON /GVARS / GVARS(100)
C
      DIMENSION AUTOPC(180)
      DIMENSION AUXVC1(4)
      DIMENSION AUXVC2(23)
      DIMENSION CONICC(29)
```

```
      DIMENSION CCON1(10)
      DIMENSION CCON2(10)
      DIMENSION CCON3(10)
      DIMENSION CCON4(10)
      DIMENSION CCON5(10)
      DIMENSION CCON6(10)
      DIMENSION DPGVC1(6)
      DIMENSION DPGVC2(27)
      DIMENSION DYNVC1(14)
      DIMENSION HOLIN1(43)
      DIMENSION MNMML1(61)
      DIMENSION MOTBL1(184)
      DIMENSION MOTBL6(122)
      DIMENSION MOTBL7(172)
      DIMENSION RANGEC(8)
      DIMENSION RMOTIC(38)
      DIMENSION RMOTI2(5)
      DIMENSION RMOTV1(253)
      DIMENSION RMOTV2(59)
      DIMENSION MOTVC1(62)
      DIMENSION MOTVC2(  9)
      DIMENSION MOTVC3(100)
      DIMENSION SAUTO1(52)
      DIMENSION SPECA1(24)
      DIMENSION GUIDI1(21)
      DIMENSION GUIDI2 (10)
      DIMENSION GUIDI3(8)
      DIMENSION GUIDV1(80)
      DIMENSION LOCAL1(137)
C
      EQUIVALENCE (AUTOPC(1),AAFFP(1))
      EQUIVALENCE (AUXVC1(1),ALPTOT    )
      EQUIVALENCE (AUXVC2(1),YXMN(1)   )
      EQUIVALENCE (CONICC(1),SEMJAX(1))
      EQUIVALENCE (CCON1(1),CON1(1,1))
      EQUIVALENCE (CCON2(1),CON1(1,2))
      EQUIVALENCE (CCON3(1),CON1(1,3))
      EQUIVALENCE (CCON4(1),CON1(1,4))
      EQUIVALENCE (CCON5(1),CON1(1,5))
      EQUIVALENCE (CCON6(1),CON1(1,6))
      EQUIVALENCE (DPGVC1(1),ALPHA     )
      EQUIVALENCE (DPGVC2(1),ROLI      )
      EQUIVALENCE (DYNVC1(1),DTIMR(1)  )
      EQUIVALENCE (HOLIN1(1),ALPARG    )
      EQUIVALENCE  (MNMML1(1),CADANM)
      EQUIVALENCE (MOTBL1(1),CST(1)    )
      EQUIVALENCE (MOTBL6(1),IXXT(1)   )
      EQUIVALENCE (MOTBL7(1),AYBCT(1))
      EQUIVALENCE (RANGEC(1),CRRNG     )
      EQUIVALENCE (RMOTIC(1),DREFR(1))
      EQUIVALENCE (RMOTI2(1),IAEROM)
```

```
      EQUIVALENCE  (RMOTV1(1),IXX    )
      EQUIVALENCE (RMOTV2(1),PND(1))
      EQUIVALENCE (MOTVC1(1),AMXB(1)   )
      EQUIVALENCE (MOTVC2(1),LONGI     )
      EQUIVALENCE (MOTVC3(1),DFNS      )
      EQUIVALENCE (SAUTO1(1),HAAP(1))
      EQUIVALENCE (SPECA1(1),SPECI(1) )
      EQUIVALENCE (GUIDI1(1),TIMEG)
      EQUIVALENCE (GUIDI2(1),YAWPK(1))
      EQUIVALENCE (GUIDI3(1),GTIME2)
      EQUIVALENCE (GUIDV1(1),GDT)
      EQUIVALENCE (LOCAL1(1),CALPHA    )
C
      DATA PRO    /502*0/
      DATA IGFLAG /10*0/
C
C     AXCALC
C
      DATA INUM /5, 24, 6, 28, 2, 4, 0, 0, 0, 0/
      DATA NSETS /6/
      DATA NVAR /69/
C

      DATA GVARS  /100*0/
C
      DATA IV(2) /0/
C     AUTOPC
      DATA AUTOPC /180*0/
C
C
C     AUXVC
C
      DATA AUXVC1/4*0/
      DATA XMAX  /10*-1.0E10/
      DATA XMIN  /10* 1.0E10/
      DATA AUXVC2/23*0/
C
C     CONICC
C
      DATA CONICC/29*0.0         /
C
C     CCONA1
C
      DATA NEQ1 /6/
      DATA HDT /0/
      DATA CCON1 /2B,4*0,.05,4*0/
      DATA CCON2 /2B,4*0,.05,4*0/
      DATA CCON3 /2B,4*0,1.0,4*0/
      DATA CCON4 /2B,4*0,.05,4*0/
      DATA CCON5 /2B,4*0,.05,4*0/
      DATA CCON6 /2B,4*0,.05,4*0/
```

```
C
C     CYCVC
C
      DATA DELT  /0.0            /
      DATA DT    /1.0     /
      DATA DTIME /1.0     /
      DATA DTM   /1.0            /
      DATA DTO   /0.0            /
      DATA ENOIS /1.E-8 /
      DATA TREF  /0.0            /
      DATA IDTAB /6*0            /
      DATA IFLG  /0              /
      DATA CYCF  /0              /
      DATA DELTT /0.0     /
      DATA GTIME /0.0     /
      DATA ATIME /0.0     /
      DATA DTG   /1.E+300/
      DATA DTA   /1.E+300/
C
C     DPGVC
C
      DATA DPGVC1/6*0/
      DATA AB      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
      DATA GB      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
      DATA IA      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
      DATA IB      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
      DATA IG      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
      DATA IL      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
      DATA LB      /1.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0/
                                DATA IGUID
     1/0     ,0     ,0     ,0     ,1     ,0     ,0     ,0     ,0     ,0
     2,0     ,2     ,1     ,0     ,0
     3,10*0
     4/
      DATA DPGVC2/27*0/
C
C     DYNVC
C
      DATA DYNVC1/14*0.0         /
C
C     HOLINC
C
      DATA HOLIN1/43*0.0         /
      DATA HOLEND /0/
C
C     INFVC
C
      DATA ESNPRT/0              /
      DATA EXTRAP,LPRNT/2*0/
      DATA PINC  /0.0            /
      DATA PRNC  /1HU            /
```

```
      DATA FID   /10HUD265 FILE,10H I.D. 0000/
      DATA INFF,IPRNTB,IPRNTR/3*0/
      DATA PSTOP /6HPSTOP /
      DATA TITLE /10*10H          /
      DATA SFID  /0   ,0   /
C
C     MNMMLT
C
      DATA ONE     /1.0/
      DATA MNMML1 /61*0/
C
C     MOTBL
C
                                DATA MOTBL1
     1/0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     2,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     3,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     4,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     5,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     6,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     7,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     8,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     9,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     0,0,1., 0,1.
     F/
                                DATA MOTBL6
     1/0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     2,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     3,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     4,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     5,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     6,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     7,0,1. /
                                DATA MOTBL7
     1/0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     2,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     3,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     4,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     5,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     6,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     7,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     8,0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     9,0,1., 0,1., 0,1., 0,1., 0,1., 0,1.
     0/
C
C     MOTIC
C
      DATA ALTREF/100.          /
      DATA LATREF/1HU           /
      DATA LONREF/0.0           /
      DATA AZREF /0.0            /
```

```
      DATA TIMREF/0.0            /
      DATA ATMOSK /1.0   ,1.0      /
      DATA AZWB   /180.           /
      DATA AZL   /0.0            /
      DATA LATL  /1HU            /
      DATA LONL  /1HU            /
      DATA CLCDMX/0.0            /
      DATA DETA  /0.0            /
      DATA DESNE /0.0            /
      DATA ETAPC /1.0,0.,0.,0.   /
      DATA ETA    /1.0/
      DATA GINT  /10*0.0         /
      DATA GO    /32.174         /
      DATA GXP,GYP,GZP/45*0.0/
      DATA HEATK /1.,17600.,26000./
      DATA ALTITO/0.0            /
      DATA ISPV  /15*1.E11       /
C     1960 FISCHER EARTH MODEL
      DATA J2    /1.0823E-3      /
      DATA J3    /0.0            /
      DATA J4    /0.0            /
      DATA LREF   /1.0            /
C     1960 FISCHER EARTH MODEL
      DATA MU    /1.4076539E+16 /
      DATA OMEGA /7.29211E-5     /
      DATA PSL    /1HU            /
      DATA PWPROP/0.0            /
      DATA RHOSL /.0023769       /
      DATA RN     /1.0            /
C     1960 FISCHER EARTH MODEL
      DATA RE     /20925741.      /
      DATA RP     /20855590.      /
      DATA SREF  /0.0            /
      DATA TSL   /1HU            /
      DATA WGTSG /1.E-10         /
      DATA WJETT,WPLD,WPROPI,WEICON/4*0.0/
      DATA XREF  /3*0.0          /
      DATA AEXP   /.64  /
      DATA CINF   /1.0  /
      DATA VINFI  /.007 /
      DATA IENGA /15*1           /
      DATA IENGT /15*1           /
      DATA IWPF  /15*0.0         /
      DATA NENG   /15       /
      DATA NFQS  /3*0.0          /
                              DATA NPC
     1/0     ,1     ,4     ,2     ,2     ,0     ,0     ,1     ,0     ,0
     2,0     ,0     ,0     ,0     ,0     ,0     ,0     ,0     ,1     ,0
     3,16*0
     4/
      DATA CPNXRD/0/
```

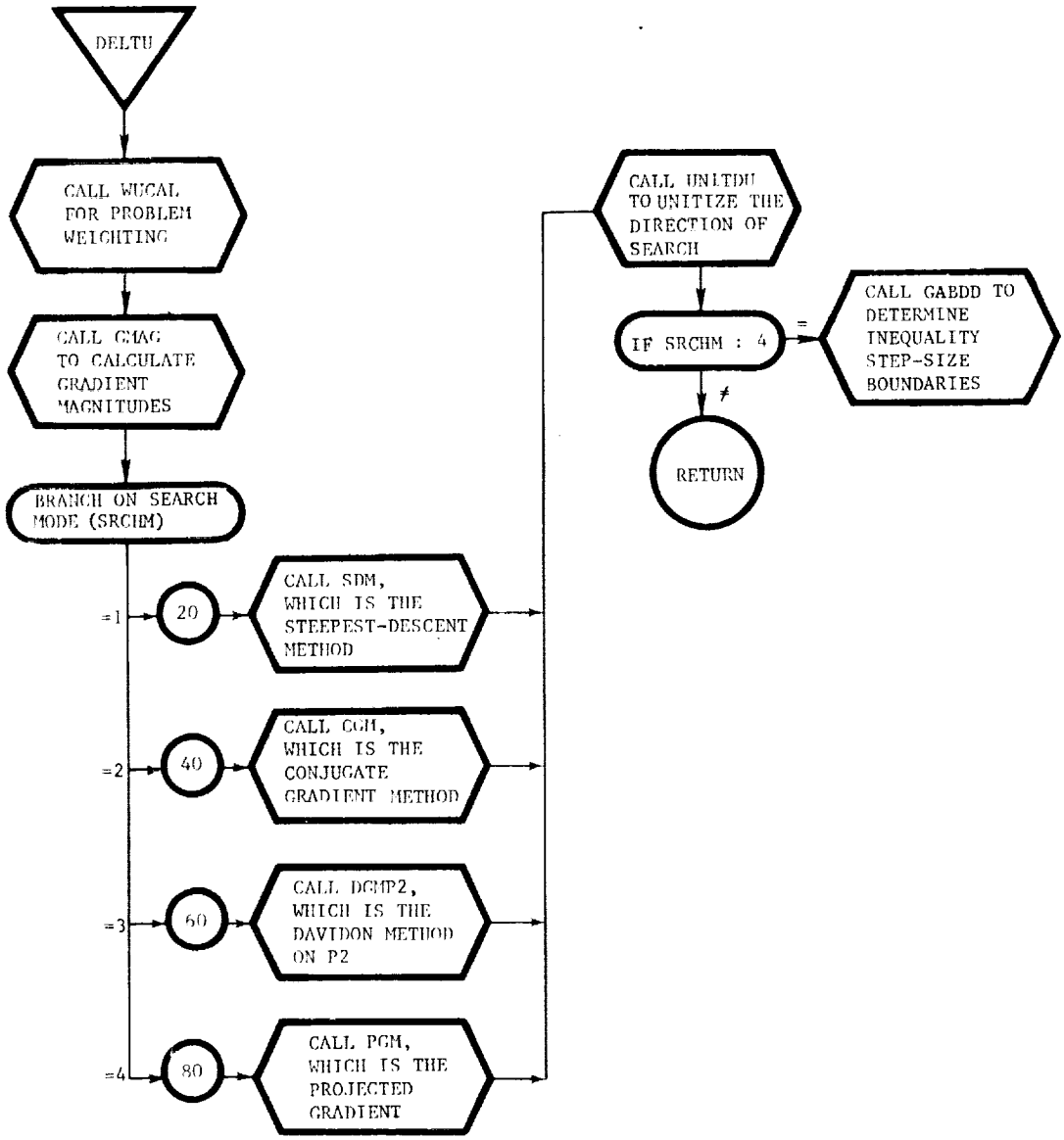
```
      DATA CPYXRD/0/
C
C     RMOTIC
C
      DATA HMAMX / 1.0E20/
      DATA HMEMX / 1.0E20/
      DATA HMRMX / 1.0E20/
      DATA AREFL /0/
      DATA EREFL /0/
      DATA RREFL /0/
      DATA DAMAX / 1.0E20/
      DATA DEMAX / 1.0E20/
      DATA DPMAX / 1.0E20/
      DATA DADMAX /1.0E20/
      DATA DEDMAX /1.0E20/
      DATA DRDMAX /1.0E20/
      DATA DAMIN /-1.0E20/
      DATA DEMIN /-1.0E20/
      DATA DPMIN /-1.0E20/
      DATA DELAAC /0/
      DATA DELEAC /0/
      DATA DELRAC /0/
      DATA DFPMAX /15*1.0E20/
      DATA DEYMAX /15*1.0E20/
      DATA RMOTIC /38*0/
      DATA FHMAX /3*1.0E20/
      DATA RMOTI2 /5*0/
C
C     MOTVC
C
      DATA MOTVC1/62*0/
      DATA GCLAT /1HU              /
      DATA GDLAT /0                /
      DATA LONG  /1HU              /
      DATA MOTVC2/9*0              /
      DATA GCRAD /1HU              /
      DATA MOTVC3/100*0/
      DATA ISV   / 1       /
      DATA REYNO /0/
      DATA VMU   /0/
      DATA CDA /0        /
      DATA CDO /0        /
      DATA CLA /0        /
      DATA CLO /0        /
      DATA CAA /0         /
      DATA CAO /0         /
      DATA CMA /0         /
      DATA CMO /0         /
      DATA CNA /0         /
      DATA CNO /0         /
      DATA CWB /0         /
```

```
      DATA CWO /0          /
      DATA CYB /0          /
      DATA CYO /0          /
C
C     RMOTVC
C
      DATA RMOTV1 /253*0/
      DATA ENGROL /15*1.0E10/
      DATA RMOTV2 /59*0/
C
C     PHZVC
C
      DATA ALTMAX/1.E20     /
      DATA ALTMIN/-5000.   /
      DATA MAXTIM/1.0E10   /
      DATA EVTF  /0                /
      DATA FESN  /100              /
      DATA IESN,PHZF,PIF,I4/4*0/
C
C     SAUTOC
C
      DATA SAUTO1 /52*0/
C
C     SPECAL
C
      DATA SPECA1/24*0             /
C
C     TGOVC
C
      DATA FUXN  /10*0             /
      DATA PCTGO /    .9 /
      DATA SAVE  /70*0             /
      DATA TGO   /10.0E10/
      DATA TIMX,ESN/2*0/
      DATA IFVNT /10*0             /
      DATA ISZEV /0                /
      DATA NXEVT /3       /
      DATA I5     /0               /
      DATA DVALUE /0/
      DATA SAVEI /40*0/
C
C
C     GUIDIC
C
      DATA GUIDI1 /21*0/
      DATA ROLIDL /1.0E20/
      DATA PITIDL /1.0E20/
      DATA YAWIDL /1.0E20/
      DATA ROLINL /-1.0E20/
      DATA PITINL /-1.0E20/
      DATA YAWINL /-1.0E20/
```

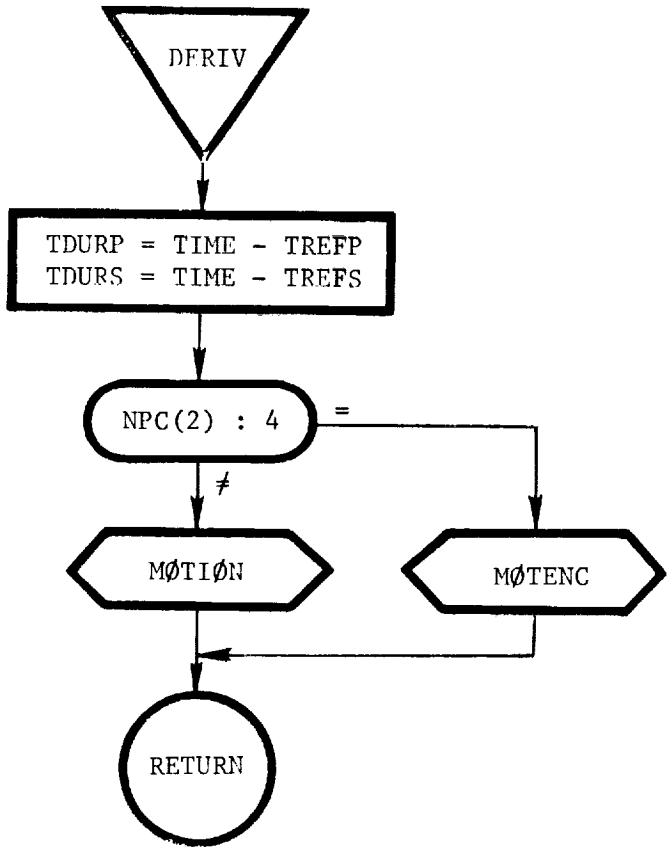
```
      DATA ROLIPL /1.0E20/
      DATA PITIPL /1.0E20/
      DATA YAWIPL /1.0E20/
      DATA GUIDI2 /10*0/
      DATA RAZD /-44.0/
      DATA GUIDI3 /8*0/
C
C     GUIDVC
C
      DATA GUIDV1 /80*0/
C
C     DYTEM
C
      DATA NMAX  /40              /
C
C     LOCAL
C
      DATA LOCAL1/137*0/
      DATA ATIMEH /-10.0E10/
      DATA SFILTR /60*0/
C
C     NPC(1) FLAGS CONIC CALCULATION
C     NPC(2) FLAGS INTEGRATION SCHEME
C     NPC(3) FLAGS INITIAL VELOCITY INPUT OPTION
C     NPC(4) FLAGS INITIAL POSITION INPUT OPTION
C     NPC(5) FLAGS ATMOSPHERE MODEL
C     NPC(6) FLAGS ATMOSPHERE WINDS
C     NPC(7) NOT USED
C     NPC(8) FLAGS AERODYNAMIC COEFFICIENT TYPE
C     NPC(9) FLAGS ENGINE TYPE
C     NPC(10) NOT USED
C     NPC(11) FLAGS ENVIRONMENTAL INEQUALITY CONSTRAINTS
C     NPC(12) FLAGS CROSS AND DOWNRANGE CALCULATION
C     NPC(13) FLAGS PROPELLANT JETTISON OPTION
C     NPC(14) FLAGS HOLD DOWN OPTION
C     NPC(15) FLAGS HEATING RATE
C     NPC(16) FLAGS EARTH MODEL
C     NPC(17) FLAGS MASS FRACTION JETTISON OPTION
C     NPC(18) FLAGS TRAJECTORY TERMINATION
C     NPC(19) FLAGS INPUT CONDITIONS PRINTOUT
C     NPC(20) FLAGS DT MODEL
C     NPC(21) FLAGS FLOWRATE METHOD FOR ROCKET ENGINES
C     NPC(22) FLAGS THROTTLING PARAMETER
C     NPC(23) NOT USED
C     NPC(24) FLAGS GENERAL INTEGRATION
C     NPC(25) NOT USED
C     NPC(26) NOT USED
C     NPC(27) FLAGS INDIVIDUAL ENGINE FLOWRATES
C     NPC(28) NOT USED
C     NPC(29) NOT USED
C     NPC(30) FLAGS WEIGHT AS FUNCTION OF TABLE LOOKUP
```

```
C       NPC(31) NOT USED
C       NPC(32)-NPC(34) NOT USED
C       NPC(35) FLAGS DIALOG GENERATED DATA INPUT
C
        END
```

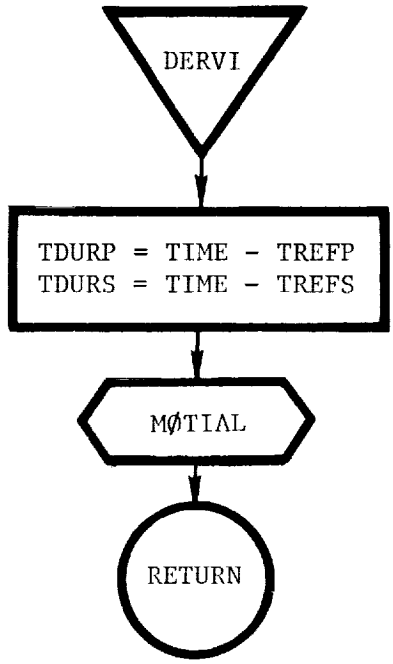
DELTU: This is the main program of overlay (2,5). This routine determines the direction of search, based on the search/optimization mode selected by user input.



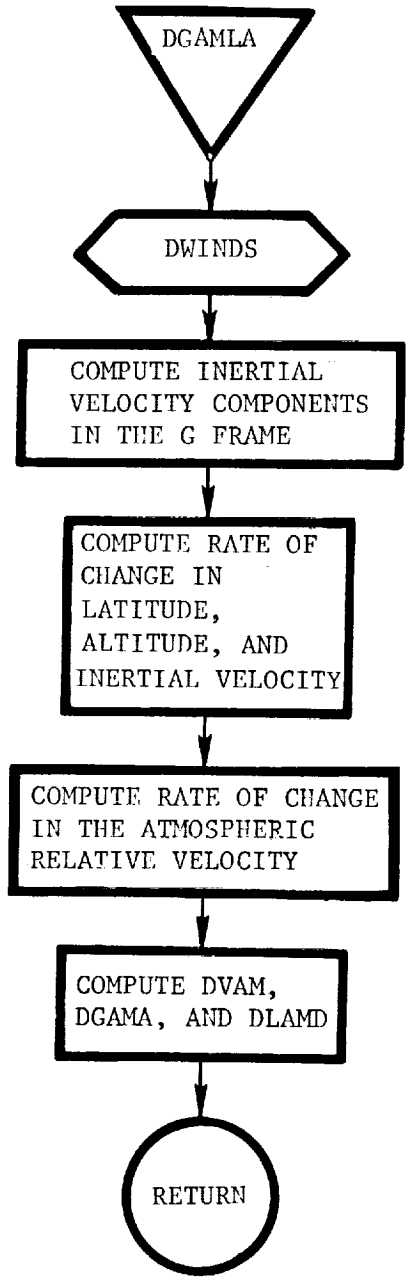
DERIV: This routine updates the time references and calls the computational routines.



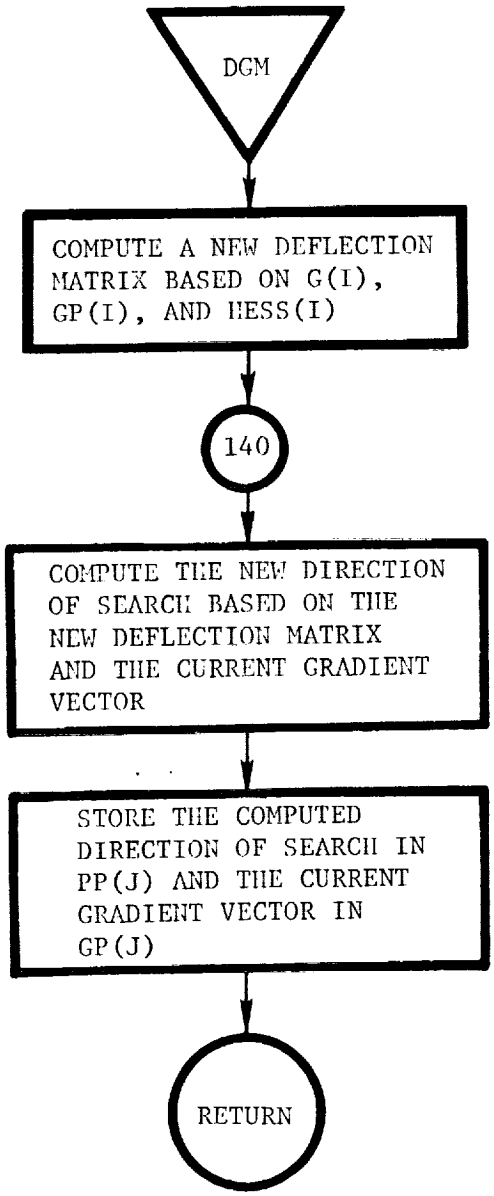
DERVI: This routine initializes the time references and calls the routines to initialize the equations of motion.



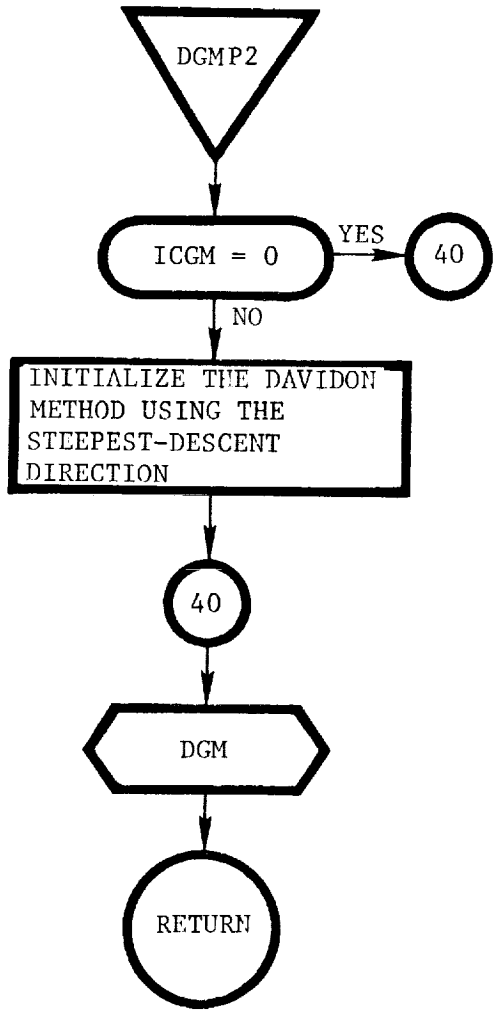
DGAMLA: This routine computes the rate of change in path angle and azimuth relative to the atmosphere.



DGM (G): This routine computes the direction of search based on the Davidon deflected gradient method and is used to minimize a scalar-valued function whose gradient vector is given by G(I).



DGMP2: This routine computes the direction of search for minimizing P2 via the Davidon variable metric method.



DICT: This routine is not call explicity, but maps the variable names to core locations at the time overlay (1,0) is called.

```
*DECK,DICT
      SUBROUTINE DICT
C*** DICT
C                DICT - DEFINES COMPUTATIONAL COMMONS
C                (IV - END) DICTIONARY VALUES DATA
C*** THE DATA STATEMENTS IN THIS ROUTINE ARE STANDARDIZED
C*** THERE MUST BE AN EVEN NUMBER OF INTEGERS IN COMMONS BETWEEN
C    IV AND END FOR THE UNIVAC 1108 DOUBLE PRECISION VERSION
C
      COMMON/IV/ IV(2)
C
C       C O M P U T A T I O N A L   D A T A   R E G I O N
C
*CALL AUTOPC
*CALL AUXVC
*CALL CONICC
*CALL CYCVC
*CALL CCONA1
*CALL DPGVC
*CALL DYNVC
*CALL INFVC
*CALL MNMMLT
*CALL HOLINC
*CALL MOTBL
*CALL MOTIC
*CALL RMOTIC
*CALL MOTVC
*CALL RMOTVC
*CALL PHZVC
*CALL RANGEC
*CALL SAUTOC
*CALL SPECAL
*CALL TGOVC
*CALL GUIDIC
*CALL GUIDVC
      COMMON/END/END
C
C***   C O M M O N S   N O T   I N C L U D E D   I N   D I C T I O N A R Y
C
*CALL DYTEM
C
C***   C O M P U T A T I O N A L   D A T A   D I C T I O N A R Y
C
      DIMENSION AUTOP1(86)
      DIMENSION AUTOP2(94)
      DIMENSION AUXVC1(47)
      DIMENSION CONICC(29)
      DIMENSION CCON1(10)
      DIMENSION CCON2(10)
      DIMENSION CCON3(10)
      DIMENSION CCON4(10)
```

```
      DIMENSION CCON5(10)
      DIMENSION CCON6(10)
      DIMENSION CYCVC1(20)
      DIMENSION DPGVC1(121)
      DIMENSION DYNVC1(14)
      DIMENSION INFVC1(23)
      DIMENSION MNMML1(62)
      DIMENSION HOLIN1(43)
      DIMENSION MOTBL6(122)
      DIMENSION MOTBL7(84)
      DIMENSION RANGEC(8)
      DIMENSION RMOTV1(126)
      DIMENSION RMOTV2(126)
      DIMENSION RMOTV3(30)
      DIMENSION RMOTV4(29)
      DIMENSION MOTIC1(126)
      DIMENSION MOTIC2(80)
      DIMENSION RMOTIC(94)
      DIMENSION MOTVC1(126)
      DIMENSION MOTVC2(66)
      DIMENSION PHZVC1(9)
      DIMENSION SAUTO1(52)
      DIMENSION SPECA1(24)
      DIMENSION TGOVC1(126)
      DIMENSION TGOVC2(12)
      DIMENSION GUIDI1(49)
      DIMENSION GUIDV1(80)
C
      EQUIVALENCE (AUTOP1(1),AAFFP(1))
      EQUIVALENCE (AUTOP2(1),ALPERR(1))
      EQUIVALENCE (AUXVC1(1),ALPTOT    )
      EQUIVALENCE (CONICC(1),SEMJAX(1))
      EQUIVALENCE (CCON1(1),CON1(1,1))
      EQUIVALENCE (CCON2(1),CON1(1,2))
      EQUIVALENCE (CCON3(1),CON1(1,3))
      EQUIVALENCE (CCON4(1),CON1(1,4))
      EQUIVALENCE (CCON5(1),CON1(1,5))
      EQUIVALENCE (CCON6(1),CON1(1,6))
      EQUIVALENCE (CYCVC1(1),DELT      )
      EQUIVALENCE (DPGVC1(1),ALPHA     )
      EQUIVALENCE (DYNVC1(1),DTIMR(1)  )
      EQUIVALENCE (INFVC1(1),ESNPRT    )
      EQUIVALENCE (MNMML1(1),ONE       )
      EQUIVALENCE (HOLIN1(1),ALPARG    )
      EQUIVALENCE (MOTBL6(1),IXXT(1))
      EQUIVALENCE (MOTBL7(1),AYBCT(1))
      EQUIVALENCE (RANGEC(1),CRRNG)
      EQUIVALENCE (RMOTV1(1),IXX(1))
      EQUIVALENCE (RMOTV2(1),DEY(8))
      EQUIVALENCE (RMOTV3(1),PND(1))
      EQUIVALENCE (RMOTV4(1),CADA(1))
```

```
      EQUIVALENCE (MOTIC1(1),ALTREF)
      EQUIVALENCE (MOTIC2(1),IENGA(8))
      EQUIVALENCE (RMOTIC(1),HMAMX(1))
      EQUIVALENCE (MOTVC1(1),AMXB(1))
      EQUIVALENCE (MOTVC2(1),XI(3))
      EQUIVALENCE (PHZVC1(1),ALTMAX    )
      EQUIVALENCE (SAUTO1(1),HAAP(1))
      EQUIVALENCE (SPECA1(1),SPECI(1) )
      EQUIVALENCE (TGOVC1(1),FUXN(1)  )
      EQUIVALENCE (TGOVC2(1),SAVEI(29))
      EQUIVALENCE (GUIDI1(1),TIMEG)
      EQUIVALENCE (GUIDV1(1),GDT)
C
C
C
                                DATA AUTOP1
     1 /6HAAFFP1,6HAAFFP2,6HAAFFP3,6HAAFFY1,6HAAFFY2,6HAAFFY3,6HACCELP
     2 ,6HACCELY,6HAFFP1 ,6HAFFP2 ,6HAFFP3 ,6HAFFY1 ,6HAFFY2 ,6HAFFY3
     3 ,6HANGAP ,6HANGAY ,6HAPRLIM,6HAPPLIM,6HAPYLIM,6HASFFP1,6HASFFP2
     4 ,6HASFFP3,6HASFFY1,6HASFFY2,6HASFFY3,6HAXBM  ,6HAYBM  ,6HAZBM
     5 ,6HAYBC  ,6HAZBC  ,6HDFFR1 ,6HDFFR2 ,6HDFFR3 ,6HCKAAP ,6HCKAAY
     6 ,6HCKAP  ,6HCKAY  ,6HCKASP ,6HCKASY ,6HCKLAP ,6HCKLVP ,6HCKLAY
     7 ,6HCKLVY ,6HDFFP1 ,6HDFFP2 ,6HDFFP3 ,6HDFFY1 ,6HDFFY2 ,6HDFFY3
     8 ,6HDISR  ,6HDISP  ,6HDISY  ,6HDSPINT,6HDSYINT,6HLDRLF ,6HPACCER
     9 ,6HYACCER,6HPLDRL ,6HYLDRL ,6HRATR  ,6HRATP  ,6HRATY  ,6HROLBER
     0 ,6HPITBER,6HYAWBER,6HRBDC  ,6HPBDC  ,6HYBDC  ,6HROLAC ,6HPITAC
     A ,6HYAWAC ,6HRFFR1 ,6HRFFR2 ,6HRFFR3 ,6HRFFP1 ,6HRFFP2 ,6HRFFP3
     B ,6HRFFY1 ,6HRFFY2 ,6HRFFY3 ,6HRRATER,6HPRATER,6HYRATER,6HTAUINT
     C ,6HTAULP ,6HTAULY
     I /
                                DATA AUTOP2
     1 /6HALPERR,6HBETERR,6HBNKERR,6HALPERI,6HALPHAC,6HAXRB  ,6HAYRB
     2 ,6HAZRB  ,6HBETAFH,6HBETAH ,6HBETAH2,6HBETDH2,6HBETDFH,6HBETDTH
     3 ,6HBLLH  ,6HCHA   ,6HCHE   ,6HCHR   ,6HCKA   ,6HCKE   ,6HCKR
     4 ,6HDADMX ,6HDEDMX ,6HDRDMX ,6HDAXB  ,6HDAYB  ,6HDAZB  ,6HDELAC
     5 ,6HDELEC ,6HDELRC ,6HDELECI,6HDELAD ,6HDELED ,6HDELRD ,6HEA2
     6 ,6HEA2I  ,6HEDELAH,6HEP1H  ,6HEY5   ,6HEY5H  ,6HEY5I  ,6HHMA
     7 ,6HHME   ,6HHMR   ,6HHMF1  ,6HHMF2  ,6HHMF3  ,6HNNPJ1 ,6HNNRJ1
     8 ,6HNNRJ2 ,6HNPPJ1 ,6HNPRJ1 ,6HNPRJ2 ,6HNNYJ1 ,6HNNYJ2 ,6HNPYJ1
     9 ,6HNPYJ2 ,6HNUDCFL,6HROLIER,6HYAWIER,6HPITIER,6HROLN  ,6HPITN
     0 ,6HYAWN  ,6HROLP  ,6HPITP  ,6HYAWP  ,6HRLLH  ,6HPITBDH,6HYAWBDH
     A ,6HTONT  ,6HTONTD ,6HTROLN ,6HTPITN ,6HTYAWN ,6HTROLND,6HTPITND
     B ,6HTYAWND,6HTROLP ,6HTPITP ,6HTYAWP ,6HTROLPD,6HTPITPD,6HTYAWPD
     C ,6HTHRSJ ,6HTISPJ ,6HUDCTR ,6HWDOTJ ,6HWPCONJ,6HYAWRER,6HPITRER
     D ,6HROLRER,6HDSPD  ,6HDSYD
     I /
C
C     AUXVC
C
                                DATA AUXVC1
     1 /6HALPTOT,6HDRAG  ,6HLIFT  ,6HDGENV ,6HXMAX1 ,6HXMAX2 ,6HXMAX3
```

```
     2 ,6HXMAX4 ,6HXMAX5 ,6HXMAX6 ,6HXMAX7 ,6HXMAX8 ,6HXMAX9 ,6HXMAX10
     3 ,6HXMIN1 ,6HXMIN2 ,6HXMIN3 ,6HXMIN4 ,6HXMIN5 ,6HXMIN6 ,6HXMIN7
     4 ,6HXMIN8 ,6HXMIN9 ,6HXMIN10,6HYXMN1 ,6HYXMN2 ,6HYXMN3 ,6HYXMN4
     5 ,6HYXMN5 ,6HYXMN6 ,6HYXMN7 ,6HYXMN8 ,6HYXMN9 ,6HYXMN10,6HYXMX1
     6 ,6HYXMX2 ,6HYXMX3 ,6HYXMX4 ,6HYXMX5 ,6HYXMX6 ,6HYXMX7 ,6HYXMX8
     7 ,6HYXMX9 ,6HYXMX10,6HXR    ,6HYR    ,6HZR
     8 /
      DATA CALNAM /73*6HCALNAM/
C
C     CONICC
C
                                DATA CONICC
     1 /6HSEMJAX,6HECCEN ,6HINC   ,6HLAN   ,6HARGP  ,6HTRUAN ,6HALTA
     2 ,6HALTP  ,6HANGMOM,6HAPORAD,6HAPVEL ,6HARGV  ,6HDECLIN,6HDVCIR
     3 ,6HECCAN ,6HENERGY,6HHYPVEL,6HLANVE ,6HMEAAN ,6HPERIOD,6HPGCLAT
     4 ,6HPGERAD,6HPGLON ,6HPGVEL ,6HRTASC ,6HTIMSP ,6HTIMTP ,6HTRUNMX
     5 ,6HVCIRC
     6 /
C
C
C     CCONA1
C
      DATA HDT /6HHDT   /
      DATA NEQ1 /6HNEQ1  /
      DATA CCON1 /6HCON11 ,6HCON21 ,6HCON31 ,6HCON41 ,6HCON51
     1           ,6HCON61 ,6HCON71 ,6HCON81 ,6HCON91 ,6HCON101
     2           /
      DATA CCON2 /6HCON12 ,6HCON22 ,6HCON32 ,6HCON42 ,6HCON52
     1           ,6HCON62 ,6HCON72 ,6HCON82 ,6HCON92 ,6HCON102
     2           /
      DATA CCON3 /6HCON13 ,6HCON23 ,6HCON33 ,6HCON43 ,6HCON53
     1           ,6HCON63 ,6HCON73 ,6HCON83 ,6HCON93 ,6HCON103
     2           /
      DATA CCON4 /6HCON14 ,6HCON24 ,6HCON34 ,6HCON44 ,6HCON54
     1           ,6HCON64 ,6HCON74 ,6HCON84 ,6HCON94 ,6HCON104
     2           /
      DATA CCON5 /6HCON15 ,6HCON25 ,6HCON35 ,6HCON45 ,6HCON55
     1           ,6HCON65 ,6HCON75 ,6HCON85 ,6HCON95 ,6HCON105
     2           /
      DATA CCON6 /6HCON16 ,6HCON26 ,6HCON36 ,6HCON46 ,6HCON56
     1           ,6HCON66 ,6HCON76 ,6HCON86 ,6HCON96 ,6HCON106
     2           /
                                DATA CYCVC1
     1 /6HDELT  ,6HDT    ,6HDTIME ,6HDTM   ,6HDTO   ,6HENOIS ,6HTREF
     2 ,6HIDTAB1,6HIDTAB2,6HIDTAB3,6HIDTAB4,6HIDTAB5,6HIDTAB6,6HIFLG
     3 ,6HCYCF  ,6HDELTT ,6HGTIME ,6HATIME ,6HDTG   ,6HDTA
     4 /
C
C
C
                                DATA DPGVC1
```

```
     1 /6HALPHA ,6HBETA   ,6HBNKANG,6HALPDOT,6HBETDOT,6HBNKDOT,6HAB1
     2 ,6HAB2   ,6HAB3   ,6HAB4   ,6HAB5   ,6HAB6   ,6HAB7   ,6HAB8
     3 ,6HAB9   ,6HGB1   ,6HGB2   ,6HGB3   ,6HGB4   ,6HGB5   ,6HGB6
     4 ,6HGB7   ,6HGB8   ,6HGB9   ,6HIA1   ,6HIA2   ,6HIA3   ,6HIA4
     5 ,6HIA5   ,6HIA6   ,6HIA7   ,6HIA8   ,6HIA9   ,6HIB11  ,6HIB12
     6 ,6HIB13  ,6HIB21  ,6HIB22  ,6HIB23  ,6HIB31  ,6HIB32  ,6HIB33
     7 ,6HIG1   ,6HIG2   ,6HIG3   ,6HIG4   ,6HIG5   ,6HIG6   ,6HIG7
     8 ,6HIG8   ,6HIG9   ,6HIL1   ,6HIL2   ,6HIL3   ,6HIL4   ,6HIL5
     9 ,6HIL6   ,6HIL7   ,6HIL8   ,6HIL9   ,6HLB1   ,6HLB2   ,6HLB3
     0 ,6HLB4   ,6HLB5   ,6HLB6   ,6HLB7   ,6HLB8   ,6HLB9   ,6HIGUID1
     A ,6HIGUID2,6HIGUID3,6HIGUID4,6HIGUID5,6HIGUID6,6HIGUID7,6HIGUID8
     B ,6HIGUID9,6HIGUID1,6HIGUID1,6HIGUID1,6HIGUID1,6HIGUID1,6HIGUID1
     C ,6HIGUID1,6HIGUID1,6HIGUID1,6HIGUID1,6HIGUID2,6HIGUID2,6HIGUID2
     D ,6HIGUID2,6HIGUID2,6HIGUID2,6HROLI  ,6HYAWI  ,6HPITI  ,6HYAWR
     E ,6HPITR  ,6HROLR  ,6HROLBD ,6HPITBD ,6HYAWBD ,6HROLID ,6HYAWID
     F ,6HPITID ,6HYAWRD ,6HPITRD ,6HROLRD ,6HALPBET,6HDE0   ,6HDE1
     G ,6HDE2   ,6HDE3   ,6HE0    ,6HE1    ,6HE2    ,6HE3    ,6HXYOME1
     H ,6HXYOME2,6HXYOME3
     I /
C
C
C
                              DATA DYNVC1
     1 /6HDTIMR1,6HDTIMR2,6HDTIMR3,6HDTIMR4,6HTDURP ,6HTIMES ,6HTIMRF1
     2 ,6HTIMRF2,6HTIMRF3,6HTIMRF4,6HTREFP ,6HTREFS ,6HNDISC ,6HNPASS
     3 /
C
C
C
                              DATA INFVC1
     1 /6HESNPRT,6HEXTRAP,6HLPRNT ,6HPINC  ,6HPRNC  ,6HFID1  ,6HFID2
     2 ,6HINFF  ,6HIPRNTB,6HIPRNTR,6HPSTOP ,6HTITLE1,6HTITLE2,6HTITLE3
     3 ,6HTITLE4,6HTITLE5,6HTITLE6,6HTITLE7,6HTITLE8,6HTITLE9,6HTITLE1
     4 ,6HSFID1 ,6HSFID2
     5/
C
C
C
                              DATA MNMML1
     1 /6HONE   ,6HCADANM,6HCADENM,6HCADRNM,6HCAF1N ,6HCAF2N ,6HCAF3N
     2 ,6HCANM  ,6HCDDANM,6HCDDENM,6HCDDRNM,6HCDF1N ,6HCDF2N ,6HCDF3N
     3 ,6HCDNM  ,6HCLDANM,6HCLDENM,6HCLDRNM,6HCLF1N ,6HCLF2N ,6HCLF3N
     4 ,6HCLNM  ,6HCLLBNM,6HCLLDAN,6HCLLDEN,6HCLLDRN,6HCLLF1N,6HCLLF2N
     5 ,6HCLLF3N,6HCLLPNM,6HCLLRNM,6HCMANM ,6HCMDANM,6HCMDENM,6HCMDRNM
     6 ,6HCMF1N ,6HCMF2N ,6HCMF3N ,6HCMQNM ,6HCNANM ,6HCNDANM,6HCNDENM
     7 ,6HCNDRNM,6HCNF1N ,6HCNF2N ,6HCNF3N ,6HCWBNM ,6HCWDANM,6HCWDENM
     8 ,6HCWDRNM,6HCWF1N ,6HCWF2N ,6HCWF3N ,6HCWPNM ,6HCWRNM ,6HCYBNM
     9 ,6HCYDANM,6HCYDENM,6HCYDRNM,6HCYF1N ,6HCYF2N ,6HCYF3N
     0 /
C
C
```

```
C
                                DATA HOLINI
     1 /6HALPARG,6HBETARG,6HBNKARG,6HETAARG,6HDGF1  ,6HDGF2  ,6HDGF3
     2 ,6HGDERV1,6HGDERV2,6HGDERV3,6HGDERV4,6HGDERV5,6HGDERV6,6HGDERV7
     3 ,6HGDERV8,6HGDERV9,6HGDERV0,6HMONF1 ,6HMONF2 ,6HMONF3 ,6HMONX1
     4 ,6HMONX2 ,6HMONX3 ,6HMONX4 ,6HMONX5 ,6HMONX6 ,6HMONX7 ,6HMONX8
     5 ,6HMONX9 ,6HMONX10,6HMONY1 ,6HMONY2 ,6HMONY3 ,6HMONY4 ,6HMONY5
     6 ,6HMONY6 ,6HMONY7 ,6HMONY8 ,6HMONY9 ,6HMONY10,6HYAWARG,6HPITARG
     7 ,6HROLARG
     8 /
C
C
C
      DATA IV(2) /6H*     /
C
      DATA CST   /6HCST   ,6HCSM   /
      DATA ATEMT /6HATEMT ,6HATEMM /
      DATA PREST /6HPREST ,6HPRESM /
      DATA VWUT  /6HVWUT  ,6HVWUM  /
      DATA VWVT  /6HVWVT  ,6HVWVM  /
      DATA VWWT  /6HVWWT  ,6HVWWM  /
      DATA AZWT  /6HAZWT  ,6HAZWM  /
      DATA VWT   /6HVWT   ,6HVWM   /
      DATA YAWT  /6HYAWT  ,6HYAWM  /
      DATA PITT  /6HPITT  ,6HPITM  /
      DATA ROLT  /6HROLT  ,6HROLM  /
      DATA CDT   /6HCDT   ,6HCDM   /
      DATA CLT   /6HCLT   ,6HCLM   /
      DATA CAT   /6HCAT   ,6HCAM   /
      DATA CNAT  /6HCNAT  ,6HCNAM  /
      DATA CYBT  /6HCYBT  ,6HCYBM  /
      DATA CMAT  /6HCMAT  ,6HCMAM  /
      DATA CWBT  /6HCWBT  ,6HCWBM  /
      DATA XCGT  /6HXCGT  ,6HXCGM  /
      DATA YCGT  /6HYCGT  ,6HYCGM  /
      DATA ZCGT  /6HZCGT  ,6HZCGM  /
                                DATA TVCIT
     1/6HTVC1T ,6HTVC1M ,6HTVC2T ,6HTVC2M ,6HTVC3T ,6HTVC3M
     2,6HTVC4T ,6HTVC4M ,6HTVC5T ,6HTVC5M ,6HTVC6T ,6HTVC6M
     3,6HTVC7T ,6HTVC7M ,6HTVC8T ,6HTVC8M ,6HTVC9T ,6HTVC9M
     4,6HTVC10T,6HTVC10M,6HTVC11T,6HTVC11M,6HTVC12T,6HTVC12M
     5,6HTVC13T,6HTVC13M,6HTVC14T,6HTVC14M,6HTVC15T,6HTVC15M
     6 /
                                DATA WDIT
     1/6HWD1T  ,6HWD1M  ,6HWD2T  ,6HWD2M  ,6HWD3T  ,6HWD3M
     2,6HWD4T  ,6HWD4M  ,6HWD5T  ,6HWD5M  ,6HWD6T  ,6HWD6M
     3,6HWD7T  ,6HWD7M  ,6HWD8T  ,6HWD8M  ,6HWD9T  ,6HWD9M
     4,6HWD10T ,6HWD10M ,6HWD11T ,6HWD11M ,6HWD12T ,6HWD12M
     5,6HWD13T ,6HWD13M ,6HWD14T ,6HWD14M ,6HWD15T ,6HWD15M
     6 /
                                DATA AEIT
```

```
     1/6HAE1T  ,6HAE1M  ,6HAE2T  ,6HAE2M  ,6HAE3T  ,6HAE3M
     2,6HAE4T  ,6HAE4M  ,6HAE5T  ,6HAE5M  ,6HAE6T  ,6HAE6M
     3,6HAE7T  ,6HAE7M  ,6HAE8T  ,6HAE8M  ,6HAE9T  ,6HAE9M
     7,6HAE10T ,6HAE10M ,6HAE11T ,6HAE11M ,6HAE12T ,6HAE12M
     5,6HAE13T ,6HAE13M ,6HAE14T ,6HAE14M ,6HAE15T ,6HAE15M
     6 /
      DATA FL1T  /6HFL1T  ,6HFL1M  /
      DATA FL2T  /6HFL2T  ,6HFL2M  /
      DATA FL3T  /6HFL3T  ,6HFL3M  /
      DATA XREFT /6HXREFT ,6HXREFM /
      DATA YREFT /6HYREFT ,6HYREFM /
      DATA ZREFT /6HZREFT ,6HZREFM /
      DATA DENST /6HDENST ,6HDENSM /
      DATA HTRTT /6HHTRTT ,6HHTRTM /
      DATA ETAT  /6HETAT  ,6HETAM  /
      DATA CAOT  /6HCAOT  ,6HCAOM  /
      DATA CNOT  /6HCNOT  ,6HCNOM  /
      DATA CYOT  /6HCYOT  ,6HCYOM  /
      DATA CDOT  /6HCDOT  ,6HCDOM  /
      DATA CLOT  /6HCLOT  ,6HCLOM  /
      DATA CWOT  /6HCWOT  ,6HCWOM  /
      DATA CMOT  /6HCMOT  ,6HCMOM  /
      DATA DENKT /6HDENKT ,6HDENKM /
      DATA GENV1T/6HGENV1T,6HGENV1M/
      DATA GENV2T/6HGENV2T,6HGENV2M/
      DATA FMASST/6HFMASST,6HFMASSM/
      DATA ZLALPT/6HZLALPT,6HZLALPM/
      DATA CAIOT /6HCAIOT ,6HCAIOM /
      DATA WGT1T /6HWGT1T ,6HWGT1M /
      DATA WGT2T /6HWGT2T ,6HWGT2M /
      DATA WGTD1T/6HWGTD1T,6HWGTD1M/
      DATA WGTD2T/6HWGTD2T,6HWGTD2M/
C
C
C
                              DATA MOTBL6
     1 /6HIXXT  ,6HIXXM  ,6HIXYT  ,6HIXYM  ,6HIXZT  ,6HIXZM  ,6HIYYT
     2 ,6HIYYM  ,6HIYZT  ,6HIYZM  ,6HIZZT  ,6HIZZM  ,6HCADAT ,6HCADAM
     3 ,6HCADET ,6HCADEM ,6HCADRT ,6HCADRM ,6HCAF1T ,6HCAF1M ,6HCAF2T
     4 ,6HCAF2M ,6HCAF3T ,6HCAF3M ,6HCDDAT ,6HCDDAM ,6HCDDET ,6HCDDEM
     5 ,6HCDDRT ,6HCDDRM ,6HCDF1T ,6HCDF1M ,6HCDF2T ,6HCDF2M ,6HCDF3T
     6 ,6HCDF3M ,6HCLDAT ,6HCLDAM ,6HCLDET ,6HCLDEM ,6HCLDRT ,6HCLDRM
     7 ,6HCLF1T ,6HCLF1M ,6HCLF2T ,6HCLF2M ,6HCLF3T ,6HCLF3M ,6HCLLBT
     8 ,6HCLLBM ,6HCLLDAT,6HCLLDAM,6HCLLDET,6HCLLDEM,6HCLLDRT,6HCLLDRM
     9 ,6HCLLF1T,6HCLLF1M,6HCLLF2T,6HCLLF2M,6HCLLF3T,6HCLLF3M,6HCLLOT
     0 ,6HCLLOM ,6HCLLPT ,6HCLLPM ,6HCLLRT ,6HCLLRM ,6HCMDAT ,6HCMDAM
     A ,6HCMDET ,6HCMDEM ,6HCMDRT ,6HCMDRM ,6HCMF1T ,6HCMF1M ,6HCMF2T
     B ,6HCMF2M ,6HCMF3T ,6HCMF3M ,6HCMOT  ,6HCMOM  ,6HCNDAT ,6HCNDAM
     C ,6HCNDET ,6HCNDEM ,6HCNDRT ,6HCNDRM ,6HCNF1T ,6HCNF1M ,6HCNF2T
     D ,6HCNF2M ,6HCNF3T ,6HCNF3M ,6HCWDAT ,6HCWDAM ,6HCWDET ,6HCWDEM
     E ,6HCWDRT ,6HCWDRM ,6HCWF1T ,6HCWF1M ,6HCWF2T ,6HCWF2M ,6HCWF3T
```

```
F ,6HCWF3M ,6HCWPT  ,6HCWPM  ,6HCWRT  ,6HCWRM  ,6HCYDAT ,6HCYDAM
G ,6HCYDET ,6HCYDEM ,6HCYDRT ,6HCYDRM ,6HCYF1T ,6HCYF1M ,6HCYF2T
H ,6HCYF2M ,6HCYF3T ,6HCYF3M
I /
                          DATA MOTBL7
1 /6HAYBCT ,6HAYBCM ,6HAZBCT ,6HAZBCM ,6HCKAT  ,6HCKAM  ,6HCKET
2 ,6HCKEM  ,6HCKRT  ,6HCKRM  ,6HCKAPT ,6HCKAPM ,6HCKAYT ,6HCKAYM
3 ,6HCKDRT ,6HCKDRM ,6HCKDPT ,6HCKDPM ,6HCKDYT ,6HCKDYM ,6HCKRRT
4 ,6HCKRRM ,6HCKRPT ,6HCKRPM ,6HCKRYT ,6HCKRYM ,6HFXBPJT,6HFXBPJM
5 ,6HFYBPJT,6HFYBPJM,6HFZBPJT,6HFZBPJM,6HCKASPT,6HCKASPM,6HCKASYT
6 ,6HCKASYM,6HRMPJT ,6HRMPJM ,6HPMPJT ,6HPMPJM ,6HYMPJT ,6HYMPJM
7 ,6HRRJIT ,6HRRJIM ,6HRYJIT ,6HRYJIM ,6HPPPJIT,6HPPPJIM,6HPNPJIT
8 ,6HPNPJIM,6HPYJIT ,6HPYJIM ,6HPRJIT ,6HPRJIM ,6HYYJIT ,6HYYJIM
9 ,6HYRJIT ,6HYRJIM ,6HKHT   ,6HKHM   ,6HKHDT  ,6HKHDM  ,6HHPT
0 ,6HHPM   ,6HDPT   ,6HDPM   ,6HRMRJT ,6HRMJRM ,6HPMRJT ,6HPMRJM
A ,6HYMRJT ,6HYMRJM ,6HCHAT  ,6HCHAM  ,6HCHET  ,6HCHEM  ,6HCHRT
B ,6HCHRM  ,6HCHF1T ,6HCHF1M ,6HCHF2T ,6HCHF2M ,6HCHF3T ,6HCHF3M
C /
C
      DATA GEN1T  /6HGEN1T ,6HGEN1M /
      DATA GEN2T  /6HGEN2T ,6HGEN2M /
      DATA GEN3T  /6HGEN3T ,6HGEN3M /
      DATA GEN4T  /6HGEN4T ,6HGEN4M /
      DATA GEN5T  /6HGEN5T ,6HGEN5M /
      DATA GEN6T  /6HGEN6T ,6HGEN6M /
      DATA GEN7T  /6HGEN7T ,6HGEN7M /
      DATA GEN8T  /6HGEN8T ,6HGEN8M /
      DATA GEN9T  /6HGEN9T ,6HGEN9M /
      DATA GEN10T /6HGEN10T,6HGEN10M/
      DATA GEN11T /6HGEN11T,6HGEN11M/
      DATA GEN12T /6HGEN12T,6HGEN12M/
      DATA GEN13T /6HGEN13T,6HGEN13M/
      DATA GEN14T /6HGEN14T,6HGEN14M/
      DATA GEN15T /6HGEN15T,6HGEN15M/
      DATA GEN16T /6HGEN16T,6HGEN16M/
      DATA GEN17T /6HGEN17T,6HGEN17M/
      DATA GEN18T /6HGEN18T,6HGEN18M/
      DATA GEN19T /6HGEN19T,6HGEN19M/
      DATA GEN20T /6HGEN20T,6HGEN20M/
      DATA GEN21T /6HGEN21T,6HGEN21M/
      DATA GEN22T /6HGEN22T,6HGEN22M/
      DATA GEN23T /6HGEN23T,6HGEN23M/
      DATA GEN24T /6HGEN24T,6HGEN24M/
      DATA GEN25T /6HGEN25T,6HGEN25M/
      DATA DELF1T/6HDELF1T,6HDELF1M/
      DATA DELF2T/6HDELF2T,6HDELF2M/
      DATA DELF3T/6HDELF3T,6HDELF3M/
      DATA RBDCT /6HRBDCT ,6HRBDCM /
      DATA PBDCT /6HPBDCT ,6HPBDCM /
      DATA YBDCT /6HYBDCT ,6HYBDCM /
      DATA ROVET /6HROVET ,6HROVEM /
```

```
      DATA CKAAPT /6HCKAAPT,6HCKAAPM/
      DATA CKAAYT /6HCKAAYT,6HCKAAYM/
      DATA CKLAYT /6HCKLAYT,6HCKLAYM/
      DATA CKLAPT /6HCKLAPT,6HCKLAPM/
      DATA CKLVPT /6HCKLVPT,6HCKLVPM/
      DATA CKLVYT /6HCKLVYT,6HCKLVYM/
      DATA CPAYRT /6HCPAYRT,6HCPAYRM/
      DATA CPAZRT /6HCPAZRT,6HCPAZRM/
      DATA CPNXRT /6HCPNXRT,6HCPNXRM/
      DATA CPNYRT /6HCPNYRT,6HCPNYRM/
      DATA CPYXRT /6HCPYXRT,6HCPYXRM/
      DATA CPYZRT /6HCPYZRT,6HCPYZRM/
C
C
C
                              DATA MOTIC1
     1 /6HALTREF,6HLATREF,6HLONREF,6HAZREF ,6HTIMREF,6HATMSK1,6HATMSK2
     2 ,6HAZWB  ,6HAZL   ,6HLATL  ,6HLONL  ,6HCLCDMX,6HDETA  ,6HDESNE
     3 ,6HETAPC1,6HETAPC2,6HETAPC3,6HETAPC4,6HETA   ,6HGINT1 ,6HGINT2
     4 ,6HGINT3 ,6HGINT4 ,6HGINT5 ,6HGINT6 ,6HGINT7 ,6HGINT8 ,6HGINT9
     5 ,6HGINT10,6HGO    ,6HGXP1  ,6HGXP2  ,6HGXP3  ,6HGXP4  ,6HGXP5
     6 ,6HGXP6  ,6HGXP7  ,6HGXP8  ,6HGXP9  ,6HGXP10 ,6HGXP11 ,6HGXP12
     7 ,6HGXP13 ,6HGXP14 ,6HGXP15 ,6HGYP1  ,6HGYP2  ,6HGYP3  ,6HGYP4
     8 ,6HGYP5  ,6HGYP6  ,6HGYP7  ,6HGYP8  ,6HGYP9  ,6HGYP10 ,6HGYP11
     9 ,6HGYP12 ,6HGYP13 ,6HGYP14 ,6HGYP15 ,6HGZP1  ,6HGZP2  ,6HGZP3
     0 ,6HGZP4  ,6HGZP5  ,6HGZP6  ,6HGZP7  ,6HGZP8  ,6HGZP9  ,6HGZP10
     A ,6HGZP11 ,6HGZP12 ,6HGZP13 ,6HGZP14 ,6HGZP15 ,6HHEATK1,6HHEATK2
     B ,6HHEATK3,6HALTITO,6HISPV1 ,6HISPV2 ,6HISPV3 ,6HISPV4 ,6HISPV5
     C ,6HISPV6 ,6HISPV7 ,6HISPV8 ,6HISPV9 ,6HISPV10,6HISPV11,6HISPV12
     D ,6HISPV13,6HISPV14,6HISPV15,6HJ2   ,6HJ3   ,6HJ4   ,6HLREF
     E ,6HMU    ,6HOMEGA ,6HPSL   ,6HPWPROP,6HRHOSL ,6HRN   ,6HRE
     F ,6HRP    ,6HSREF  ,6HTSL   ,6HWGTSG ,6HWJETT ,6HWPLD  ,6HWPROPI
     G ,6HWEICON,6HXREF  ,6HYREF  ,6HZREF  ,6HAEXP  ,6HCINF  ,6HVINFI
     H ,6HIENGA1,6HIENGA2,6HIENGA3,6HIENGA4,6HIENGA5,6HIENGA6,6HIENGA7
     I /
                              DATA MOTIC2
     1 /6HIENGA8,6HIENGA9,6HIENGA1,6HIENGA1,6HIENGA1,6HIENGA1,6HIENGA1
     2 ,6HIENGA1,6HIENGT1,6HIENGT2,6HIENGT3,6HIENGT4,6HIENGT5,6HIENGT6
     3 ,6HIENGT7,6HIENGT8,6HIENGT9,6HIENGT1,6HIENGT1,6HIENGT1,6HIENGT1
     4 ,6HIENGT1,6HIENGT1,6HIWPF1 ,6HIWPF2 ,6HIWPF3 ,6HIWPF4 ,6HIWPF5
     5 ,6HIWPF6 ,6HIWPF7 ,6HIWPF8 ,6HIWPF9 ,6HIWPF10,6HIWPF11,6HIWPF12
     6 ,6HIWPF13,6HIWPF14,6HIWPF15,6HNENG  ,6HNEQS1 ,6HNEQS2 ,6HNEQS3
     7 ,6HNPC1  ,6HNPC2  ,6HNPC3  ,6HNPC4  ,6HNPC5  ,6HNPC6  ,6HNPC7
     8 ,6HNPC8  ,6HNPC9  ,6HNPC10 ,6HNPC11 ,6HNPC12 ,6HNPC13 ,6HNPC14
     9 ,6HNPC15 ,6HNPC16 ,6HNPC17 ,6HNPC18 ,6HNPC19 ,6HNPC20 ,6HNPC21
     0 ,6HNPC22 ,6HNPC23 ,6HNPC24 ,6HNPC25 ,6HNPC26 ,6HNPC27 ,6HNPC28
     A ,6HNPC29 ,6HNPC30 ,6HNPC31 ,6HNPC32 ,6HNPC33 ,6HNPC34 ,6HNPC35
     B ,6HNPC36 ,6HCPNXRD,6HCPYXRD
     C /
C
C
```

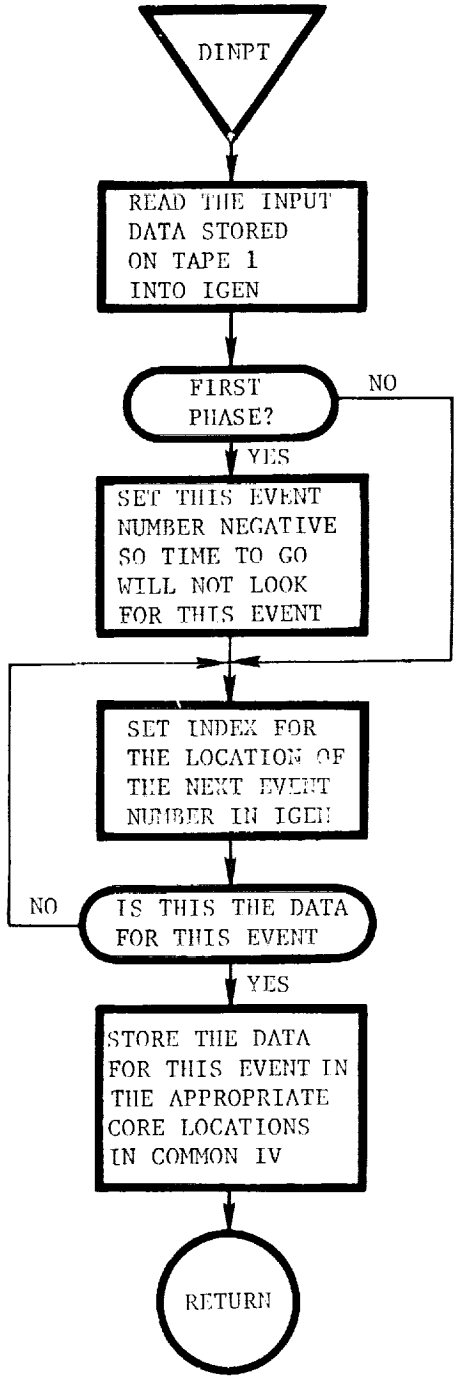
```
C
                              DATA RANGEC
     1 /6HCRRNG ,6HDPRNG1,6HDPRNG2,6HDWNRNG,6HXISAV1,6HXISAV2,6HXISAV3
     2 ,6HIRANGE
     3 /
C
C
C
                              DATA RMOTIC
     1 /6HHMAMX ,6HHMEMX ,6HHMRMX ,6HAREFL ,6HEREFL ,6HRREFL ,6HDAMAX
     2 ,6HDEMAX ,6HDRMAX ,6HDADMAX,6HDEDMAX,6HDRDMAX,6HDAMIN ,6HDEMIN
     3 ,6HDRMIN ,6HDELAAC,6HDELEAC,6HDELRAC,6HDIPM1 ,6HDIPM2 ,6HDIPM3
     4 ,6HDIPM4 ,6HDIPM5 ,6HDIPM6 ,6HDIPM7 ,6HDIPM8 ,6HDIPM9 ,6HDIPM10
     5 ,6HDIPM11,6HDIPM12,6HDIPM13,6HDIPM14,6HDIPM15,6HDIYM1 ,6HDIYM2
     6 ,6HDIYM3 ,6HDIYM4 ,6HDIYM5 ,6HDIYM6 ,6HDIYM7 ,6HDIYM8 ,6HDIYM9
     7 ,6HDIYM10,6HDIYM11,6HDIYM12,6HDIYM13,6HDIYM14,6HDIYM15,6HDREFR
     8 ,6HDREFP ,6HDREFY ,6HENGF1 ,6HENGF2 ,6HENGF3 ,6HENGF4 ,6HENGF5
     9 ,6HENGF6 ,6HENGF7 ,6HENGF8 ,6HENGF9 ,6HENGF10,6HENGF11,6HENGF12
     0 ,6HENGF13,6HENGF14,6HENGF15,6HNAUTOP,6HNGUIDF,6HRENG1 ,6HRENG2
     A ,6HRENG3 ,6HRENG4 ,6HRENG5 ,6HRENG6 ,6HRENG7 ,6HRENG8 ,6HRENG9
     B ,6HRENG10,6HRENG11,6HRENG12,6HRENG13,6HRENG14,6HRENG15,6HFREFL1
     C ,6HFREFL2,6HFREFL3,6HFHMAX1,6HFHMAX2,6HFHMAX3,6HIAEROM,6HDTHP
     D ,6HDPHP  ,6HDPSY  ,6HDPHY
     D /
                              DATA MOTVC1
     1 /6HAMXB  ,6HAMYB  ,6HAMZB  ,6HASM   ,6HASMG  ,6HATEM  ,6HAXB
     2 ,6HAYB   ,6HAZB   ,6HASXI  ,6HASYI  ,6HASZI  ,6HAXI   ,6HAYI
     3 ,6HAZI   ,6HCA    ,6HCN    ,6HCD    ,6HCL    ,6HCM    ,6HCY
     4 ,6HCW    ,6HCS    ,6HDFVAL1,6HDFVAL2,6HDFVAL3,6HDFVLH1,6HDFVLH2
     5 ,6HDFVLH3,6HGAMAD ,6HAZVAD ,6HDMASS ,6HHEATRT,6HDUA   ,6HDVA
     6 ,6HDWA   ,6HVELAD ,6HDVWH1 ,6HDVWH2 ,6HDVWH3 ,6HETAL  ,6HFAXB
     7 ,6HFAYB  ,6HFAZB  ,6HFTXB  ,6HFTYB  ,6HFTZB  ,6HFVAL1 ,6HFVAL2
     8 ,6HFVAL3 ,6HGAMMAA,6HGAMMAI,6HGAMMAR,6HGXI   ,6HGYI   ,6HGZI
     9 ,6HH     ,6HHTURB ,6HHTURBD,6HAZVELA,6HAZVELI,6HAZVELR,6HGCLAT
     0 ,6HGDLAT ,6HLONG  ,6HLONGI ,6HMACH  ,6HMASS  ,6HPJETTS,6HPRES
     A ,6HPWDOT ,6HDYNP  ,6HQALPHA,6HTLHEAT,6HGCRAD ,6HDENS  ,6HRSC
     B ,6HRS    ,6HTHRUST,6HTIME  ,6HTMXB  ,6HTMYB  ,6HTMZB  ,6HTTMXB
     C ,6HTTMYB ,6HTTMZB ,6HTVAC  ,6HU    ,6HV    ,6HW    ,6HUA
     D ,6HVA    ,6HWA    ,6HVELA  ,6HVAXI  ,6HVAYI  ,6HVAZI  ,6HUB
     E ,6HVB    ,6HWB    ,6HVELI  ,6HUR    ,6HVR    ,6HWR    ,6HVELR
     F ,6HVRXI  ,6HVRYI  ,6HVRZI  ,6HUW    ,6HVW    ,6HWW    ,6HVWXI
     G ,6HVWYI  ,6HVWZI  ,6HVXI   ,6HVYI   ,6HVZI   ,6HWDOT  ,6HWEIGHT
     H ,6HWJETTM,6HWPROP ,6HXCG   ,6HYCG   ,6HZCG   ,6HXI    ,6HYI
     I /
                              DATA MOTVC2
     1 /6HZI    ,6HDCLV  ,6HDCDV  ,6HVINV  ,6HAE1   ,6HAE2   ,6HAE3
     2 ,6HAE4   ,6HAE5   ,6HAE6   ,6HAE7   ,6HAE8   ,6HAE9   ,6HAE10
     3 ,6HAE11  ,6HAE12  ,6HAE13  ,6HAE14  ,6HAE15  ,6HWD1   ,6HWD2
     4 ,6HWD3   ,6HWD4   ,6HWD5   ,6HWD6   ,6HWD7   ,6HWD8   ,6HWD9
     5 ,6HWD10  ,6HWD11  ,6HWD12  ,6HWD13  ,6HWD14  ,6HWD15  ,6HTHR1
     6 ,6HTHR2  ,6HTHR3  ,6HTHR4  ,6HTHR5  ,6HTHR6  ,6HTHR7  ,6HTHR8
```

```
     7 ,6HTHR9   ,6HTHR10 ,6HTHR11 ,6HTHR12 ,6HTHR13 ,6HTHR14 ,6HTHR15
     8 ,6HISV    ,6HREYNO ,6HVMU   ,6HCDA   ,6HCDO   ,6HCLA   ,6HCLO
     9 ,6HCAA    ,6HCAO   ,6HCMA   ,6HCMO   ,6HCNA   ,6HCNO   ,6HCWB
     0 ,6HCWO    ,6HCYB   ,6HCYO
     A /
C
C
C
                              DATA RMOTV1
     1 /6HIXX    ,6HIXY    ,6HIXZ    ,6HIYX    ,6HIYY    ,6HIYZ    ,6HIZX
     2 ,6HIZY    ,6HIZZ    ,6HIXXD   ,6HIXYD   ,6HIXZD   ,6HIYXD   ,6HIYYD
     3 ,6HIYZD   ,6HIZXD   ,6HIZYD   ,6HIZZD   ,6HISYM6  ,6HROLB   ,6HPITB
     4 ,6HYAWB   ,6HROLBDD,6HPITBDD,6HYAWBDD,6HTHRXB1,6HTHRXB2,6HTHRXB3
     5 ,6HTHRXB4,6HTHRXB5,6HTHRXB6,6HTHRXB7,6HTHRXB8,6HTHRXB9,6HTHRX10
     6 ,6HTHRX11,6HTHRX12,6HTHRX13,6HTHRX14,6HTHRX15,6HTHRYB1,6HTHRYB2
     7 ,6HTHRYB3,6HTHRYB4,6HTHRYB5,6HTHRYB6,6HTHRYB7,6HTHRYB8,6HTHRYB9
     8 ,6HTHRY10,6HTHRY11,6HTHRY12,6HTHRY13,6HTHRY14,6HTHRY15,6HTHRZB1
     9 ,6HTHRZB2,6HTHRZB3,6HTHRZB4,6HTHRZB5,6HTHRZB6,6HTHRZB7,6HTHRZB8
     0 ,6HTHRZB9,6HTHRZ10,6HTHRZ11,6HTHRZ12,6HTHRZ13,6HTHRZ14,6HTHRZ15
     A ,6HROLBDR,6HPITBDR,6HYAWBDR,6HKRDA   ,6HKRDE   ,6HKRDR   ,6HKPDA
     B ,6HKPDE   ,6HKPDR   ,6HKYDA   ,6HKYDE   ,6HKYDR   ,6HDELA   ,6HDELE
     C ,6HDELR   ,6HDELR   ,6HDLEAO ,6HDELEO ,6HDELRO ,6HDEP1   ,6HDEP2
     D ,6HDEP3   ,6HDEP4   ,6HDEP5   ,6HDEP6   ,6HDEP7   ,6HDEP8   ,6HDEP9
     E ,6HDEP10 ,6HDEP11 ,6HDEP12 ,6HDEP13 ,6HDEP14 ,6HDEP15 ,6HDEPO1
     F ,6HDEPO2 ,6HDEPO3 ,6HDEPO4 ,6HDEPO5 ,6HDEPO6 ,6HDEPO7 ,6HDEPOR
     G ,6HDEPO9 ,6HDEPO10,6HDEPO11,6HDEPO12,6HDEPO13,6HDEPO14,6HDEPO15
     H ,6HDEY1   ,6HDEY2   ,6HDEY3   ,6HDEY4   ,6HDEY5   ,6HDEY6   ,6HDEY7
     I /
                              DATA RMOTV2
     1 /6HDEY8   ,6HDEY9   ,6HDEY10 ,6HDEY11 ,6HDEY12 ,6HDEY13 ,6HDEY14
     2 ,6HDEY15 ,6HDEYO1 ,6HDEYO2 ,6HDEYO3 ,6HDEYO4 ,6HDEYO5 ,6HDEYO6
     3 ,6HDEYO7 ,6HDEYO8 ,6HDEYO9 ,6HDEYO10,6HDEYO11,6HDEYO12,6HDEYO13
     4 ,6HDEYO14,6HDEYO15,6HKRDP1 ,6HKRDP2 ,6HKRDP3 ,6HKRDP4 ,6HKRDP5
     5 ,6HKRDP6 ,6HKRDP7 ,6HKRDP8 ,6HKRDP9 ,6HKRDP10,6HKRDP11,6HKRDP12
     6 ,6HKRDP13,6HKRDP14,6HKRDP15,6HKPDP1 ,6HKPDP2 ,6HKPDP3 ,6HKPDP4
     7 ,6HKPDP5 ,6HKPDP6 ,6HKPDP7 ,6HKPDP8 ,6HKPDP9 ,6HKPDP10,6HKPDP11
     8 ,6HKPDP12,6HKPDP13,6HKPDP14,6HKPDP15,6HKRDY1 ,6HKRDY2 ,6HKRDY3
     9 ,6HKRDY4 ,6HKRDY5 ,6HKRDY6 ,6HKRDY7 ,6HKRDY8 ,6HKRDY9 ,6HKRDY10
     0 ,6HKRDY11,6HKRDY12,6HKRDY13,6HKRDY14,6HKRDY15,6HCHDA   ,6HCHDE
     A ,6HCHDR   ,6HCDDE   ,6HCLDE   ,6HCLL    ,6HCYDR   ,6HCYDA   ,6HCYDE
     B ,6HCMDE   ,6HCWP    ,6HCWR    ,6HCLLO   ,6HCLLB   ,6HCLLDR  ,6HCLLDA
     C ,6HCLLDE  ,6HCLLR   ,6HCLLP   ,6HCDF1   ,6HCDF2   ,6HCDF3   ,6HCLF1
     D ,6HCLF2   ,6HCLF3   ,6HCAF1   ,6HCAF2   ,6HCAF3   ,6HCNF1   ,6HCNF2
     E ,6HCNF3   ,6HCYF1   ,6HCYF2   ,6HCYF3   ,6HCMF1   ,6HCMF2   ,6HCMF3
     F ,6HCWF1   ,6HCWF2   ,6HCWF3   ,6HCLLF1  ,6HCLLF2  ,6HCLLF3  ,6HENGRL1
     G ,6HENGRL2,6HENGRL3,6HENGRL4,6HENGRL5,6HENGRL6,6HENGRL7,6HENGRL8
     H ,6HENGRL9,6HENGR10,6HENGR11,6HENGR12,6HENGR13,6HENGR14,6HENGR15
     I /
                              DATA RMOTV3
     1 /6HPND    ,6HQND    ,6HRND    ,6HKYDY1 ,6HKYDY2 ,6HKYDY3 ,6HKYDY4
     2 ,6HKYDY5 ,6HKYDY6 ,6HKYDY7 ,6HKYDY8 ,6HKYDY9 ,6HKYDY10,6HKYDY11
```

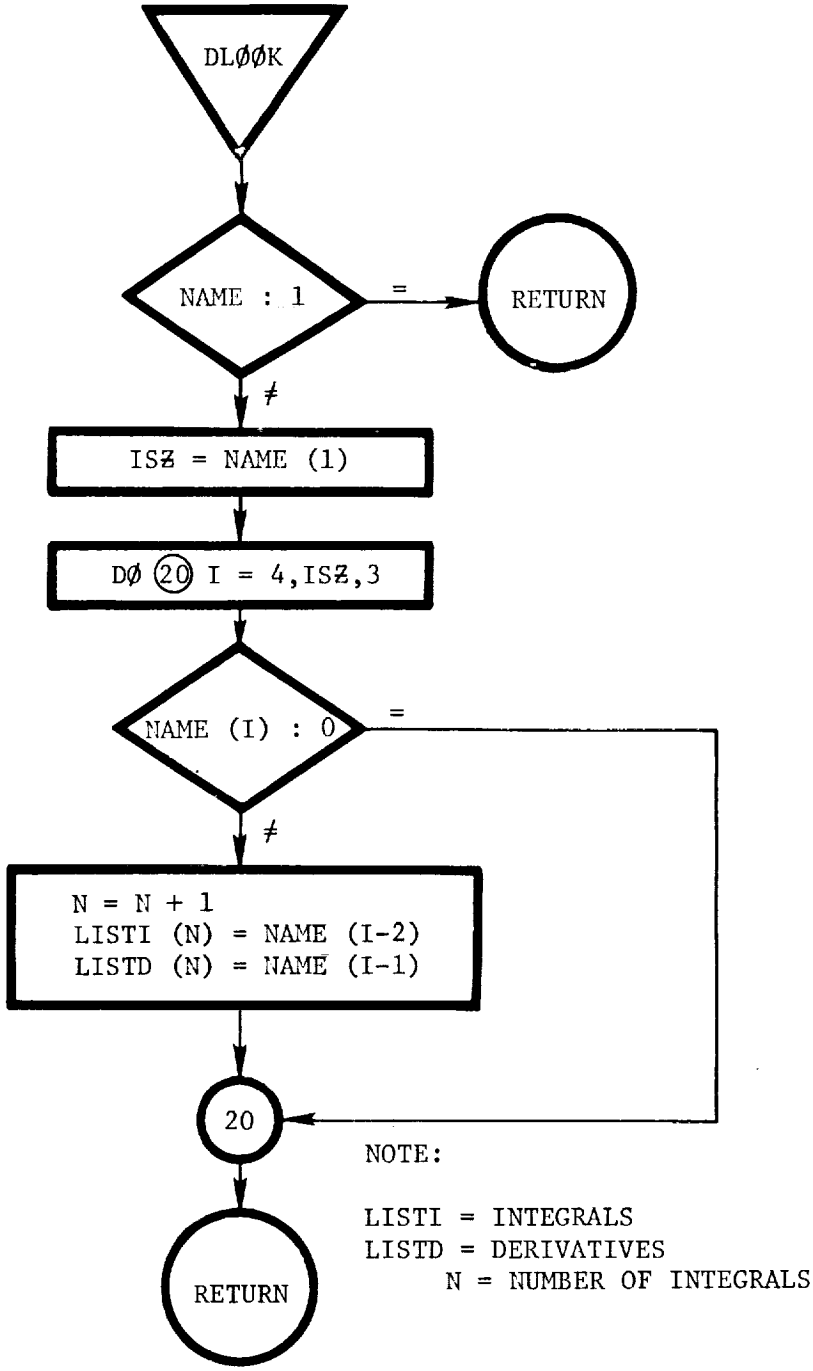
```
     3 ,6HKYDY12,6HKYDY13,6HKYDY14,6HKYDY15,6HFTTXB1,6HFTTXB2,6HFTTXB3
     4 ,6HRCSMXB,6HRCSMYB,6HRCSMZB,6HRCSFXB,6HRCSFYB,6HRCSFZB,6HDELF1
     5 ,6HDELF2 ,6HDELF3
     6 /
                                DATA RMOTV4
     1 /6HCADA  ,6HCADE  ,6HCADR  ,6HCDDA  ,6HCDDR  ,6HCLDA  ,6HCLDR
     2 ,6HCMDA  ,6HCMDR  ,6HCMO   ,6HCNDA  ,6HCNDE  ,6HCNDR  ,6HCWDA
     3 ,6HCWDE  ,6HCWDR  ,6HIAEROH,6HCPAYR ,6HCPAZR ,6HCPNXR ,6HCPNYR
     4 ,6HCPYXR ,6HCPYZR ,6HMISP1 ,6HMISP2 ,6HMISP3 ,6HMISY1 ,6HMISY2
     5 ,6HMISY3
     4 /
                                DATA PHZVC1
     1 /6HALTMAX,6HALTMIN,6HMAXTIM,6HEVTF  ,6HFESN  ,6HIESN  ,6HPHZF
     2 ,6HPIF   ,6HI4
     3 /
C
C
C
                                DATA SAUTO1
     1 /6HHAAP  ,6HHAAY  ,6HHACP1 ,6HHACP2 ,6HHACP3 ,6HHACY1 ,6HHACY2
     2 ,6HHACY3 ,6HHDAAP1,6HHDAAP2,6HHDAAP3,6HHDAAY1,6HHDAAY2,6HHDAAY3
     3 ,6HHDSPA1,6HHDSPA2,6HHDSPA3,6HHDSYA1,6HHDSYA2,6HHDSYA3,6HHDSP1
     4 ,6HHDSP2 ,6HHDSP3 ,6HHDSR1 ,6HHDSR2 ,6HHDSR3 ,6HHDSY1 ,6HHDSY2
     5 ,6HHDSY3 ,6HHLRPI ,6HHLRPO ,6HHLRYI ,6HHLRYO ,6HHPA   ,6HHPD
     6 ,6HHPR   ,6HHRD   ,6HHRR   ,6HHRTP1 ,6HHRTP2 ,6HHRTP3 ,6HHRTR1
     7 ,6HHRTR2 ,6HHRTR3 ,6HHRTY1 ,6HHRTY2 ,6HHRTY3 ,6HHSPA  ,6HHSYA
     8 ,6HHYA   ,6HHYD   ,6HHYR
     1 /
C
C
C
                                DATA SPECA1
     1 /6HSPECI1,6HSPECI2,6HSPECI3,6HSPECI4,6HSPECI5,6HSPECI6,6HSPECI7
     2 ,6HSPECI8,6HSPECI9,6HSPECV1,6HSPECV2,6HSPECV3,6HSPECV4,6HSPECV5
     3 ,6HSPECV6,6HSPECV7,6HSPECV8,6HSPECV9,6HNSPEC1,6HNSPEC2,6HNSPEC3
     4 ,6HNSPEC4,6HNSPEC5,6HNSPEC6
     5 /
C
C
C
                                DATA TGOVC1
     1 /6HFUXN1 ,6HFUXN2 ,6HFUXN3 ,6HFUXN4 ,6HFUXN5 ,6HFUXN6 ,6HFUXN7
     2 ,6HFUXN8 ,6HFUXN9 ,6HFUXN10,6HPCTGO ,6HSAVE1 ,6HSAVE2 ,6HSAVE3
     3 ,6HSAVE4 ,6HSAVE5 ,6HSAVE6 ,6HSAVE7 ,6HSAVE8 ,6HSAVE9 ,6HSAVE10
     4 ,6HSAVE11,6HSAVE12,6HSAVE13,6HSAVE14,6HSAVE15,6HSAVE16,6HSAVE17
     5 ,6HSAVE18,6HSAVE19,6HSAVE20,6HSAVE21,6HSAVE22,6HSAVE23,6HSAVE24
     6 ,6HSAVE25,6HSAVE26,6HSAVE27,6HSAVE28,6HSAVE29,6HSAVE30,6HSAVE31
     7 ,6HSAVE32,6HSAVE33,6HSAVE34,6HSAVE35,6HSAVE36,6HSAVE37,6HSAVE38
     8 ,6HSAVE39,6HSAVE40,6HSAVE41,6HSAVE42,6HSAVE43,6HSAVE44,6HSAVE45
     9 ,6HSAVE46,6HSAVE47,6HSAVE48,6HSAVE49,6HSAVE50,6HSAVE51,6HSAVE52
     0 ,6HSAVE53,6HSAVE54,6HSAVE55,6HSAVE56,6HSAVE57,6HSAVE58,6HSAVE59
```

```
     A ,6HSAVE60,6HSAVE61,6HSAVE62,6HSAVE63,6HSAVE64,6HSAVE65,6HSAVE66
     B ,6HSAVE67,6HSAVE68,6HSAVE69,6HSAVE70,6HTGO   ,6HTIMX  ,6HESN
     C ,6HIEVNT1,6HIEVNT2,6HIEVNT3,6HIEVNT4,6HIEVNT5,6HIEVNT6,6HIEVNT7
     D ,6HIEVNT8,6HIEVNT9,6HIEVN10,6HISZEV ,6HIXEVT ,6HI5    ,6HDVALUE
     E ,6HSAVEI1,6HSAVEI2,6HSAVEI3,6HSAVEI4,6HSAVEI5,6HSAVEI6,6HSAVEI7
     F ,6HSAVEI8,6HSAVEI9,6HSAVI10,6HSAVI11,6HSAVI12,6HSAVI13,6HSAVI14
     G ,6HSAVI15,6HSAVI16,6HSAVI17,6HSAVI18,6HSAVI19,6HSAVI20,6HSAVI21
     H ,6HSAVI22,6HSAVI23,6HSAVI24,6HSAVI25,6HSAVI26,6HSAVI27,6HSAVI28
     I /
                              DATA TGOVC2
     1 /6HSAVI29,6HSAVI30,6HSAVI31,6HSAVI32,6HSAVI33,6HSAVI34,6HSAVI35
     2 ,6HSAVI36,6HSAVI37,6HSAVI38,6HSAVI39,6HSAVI40
     3 /
C
C
C
C
C
C
C
                              DATA GUIDI1
     1 /6HTIMEG ,6HTIMEA ,6HGTIMF1,6HGVRI1 ,6HGVRI2 ,6HGVRI3 ,6HGVRI4
     2 ,6HGVRI5 ,6HGVRI6 ,6HGVRI7 ,6HGVRI8 ,6HGVRI9 ,6HGVRI10,6HIGF1
     3 ,6HIGF2  ,6HIGF3  ,6HIGF4  ,6HIGF5  ,6HIGF6  ,6HIGSTRT,6HIGINIT
     4 ,6HROLIDL,6HPITIDL,6HYAWIDL,6HROLINL,6HPITINL,6HYAWINL,6HROLIPL
     5 ,6HPITIPL,6HYAWIPL,6HYAWPK1,6HYAWPK2,6HROLBC ,6HPITBC ,6HYAWBC
     6 ,6HITRANS,6HISLECT,6HISTRT ,6HTLATCD,6HTLONGD,6HRAZD  ,6HGTIME2
     7 ,6HISTART,6HPITCF1,6HPITCF2,6HPITCF3,6HPITCF4,6HPITCF5,6HNPITG
     8/
                              DATA GUIDVI
     1/6HGDT   ,6HADT   ,6HGPXI  ,6HGPYI  ,6HGPZI  ,6HGPVXI ,6HGPVYI
     2,6HGPVZI ,6HGPAXI ,6HGPAYI ,6HGPAZI ,6HGALPHA,6HGBETA ,6HGBANK
     3,6HGYAWR ,6HGPITR ,6HGROLR ,6HGROLI ,6HGYAWI ,6HGPITI ,6HGASM
     4,6HGPASXI,6HGPASYI,6HGPASZI,6HGTHRST,6HGWGT  ,6HGWDOT ,6HGVRC1
     5,6HGVRC2 ,6HGVRC3 ,6HGVRC4 ,6HGVRC5 ,6HGVRC6 ,6HGVRC7 ,6HGVRC8
     6,6HGVRC9 ,6HGVRC10,6HGLONG ,6HGVELR ,6HGGCLAT,6HGAZVR ,6HGVELI
     7,6HGGAMAR,6HGALTIT,6HGGCRAD,6HGLOD  ,6HGRDOT ,6HGXR   ,6HGYR
     8,6HGZR   ,6HALPGC ,6HBETGC ,6HBNKGC ,6HROLIGC,6HYAWIGC,6HPITIGC
     9,6HRIDGC ,6HYIDGC ,6HPIDGC ,6HROLBDC,6HPITBDC,6HYAWBDC,6HALDREF
     0,6HDELAZ ,6HDRAGGC,6HDRAGRP,6HLOD1  ,6HRDTREF,6HRK2ROL,6HTRANGE
     A,6HYAWRGC,6HPITRGC,6HROLRGC,6HSLECT ,6HHP    ,6HHDP   ,6HVELRX
     B,6HVELRZ ,6HVBN   ,6HVBNS
     B/
      END
```

DINPT: This routine reads the previously stored input data from the disc and locates the data for the current phase.



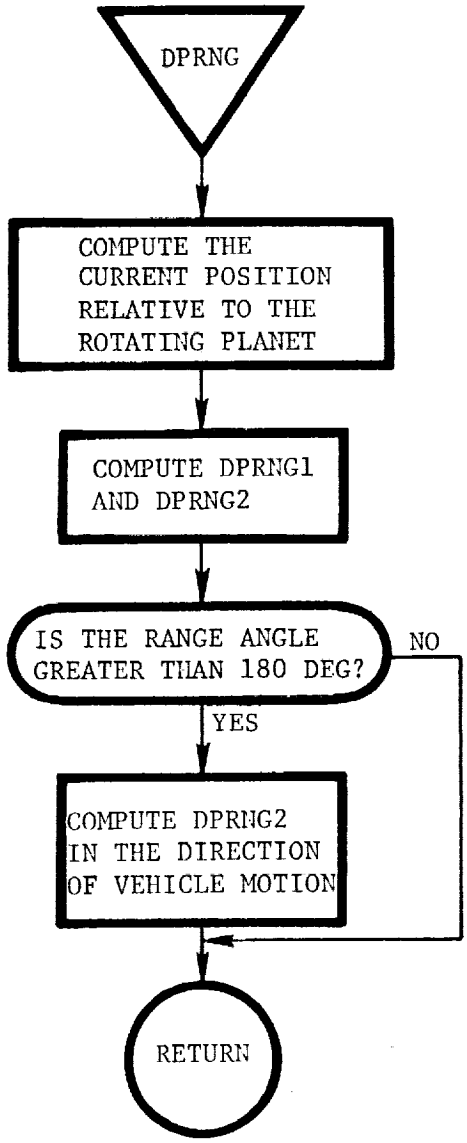
DLØØK (NAME): This routine sets the core addresses of the variables (NAME) to be integrated into the integration list.



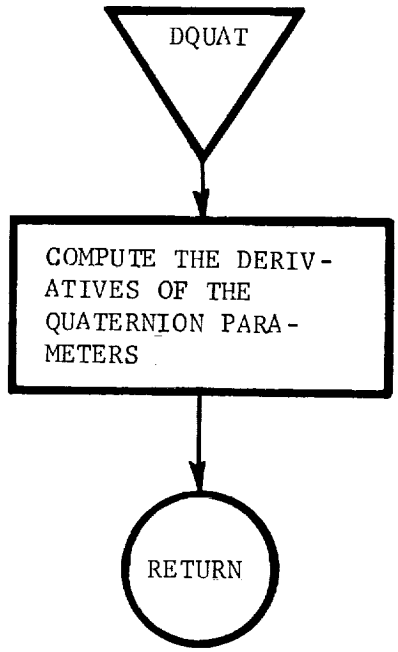
NOTE:

LISTI = INTEGRALS
LISTD = DERIVATIVES
N = NUMBER OF INTEGRALS

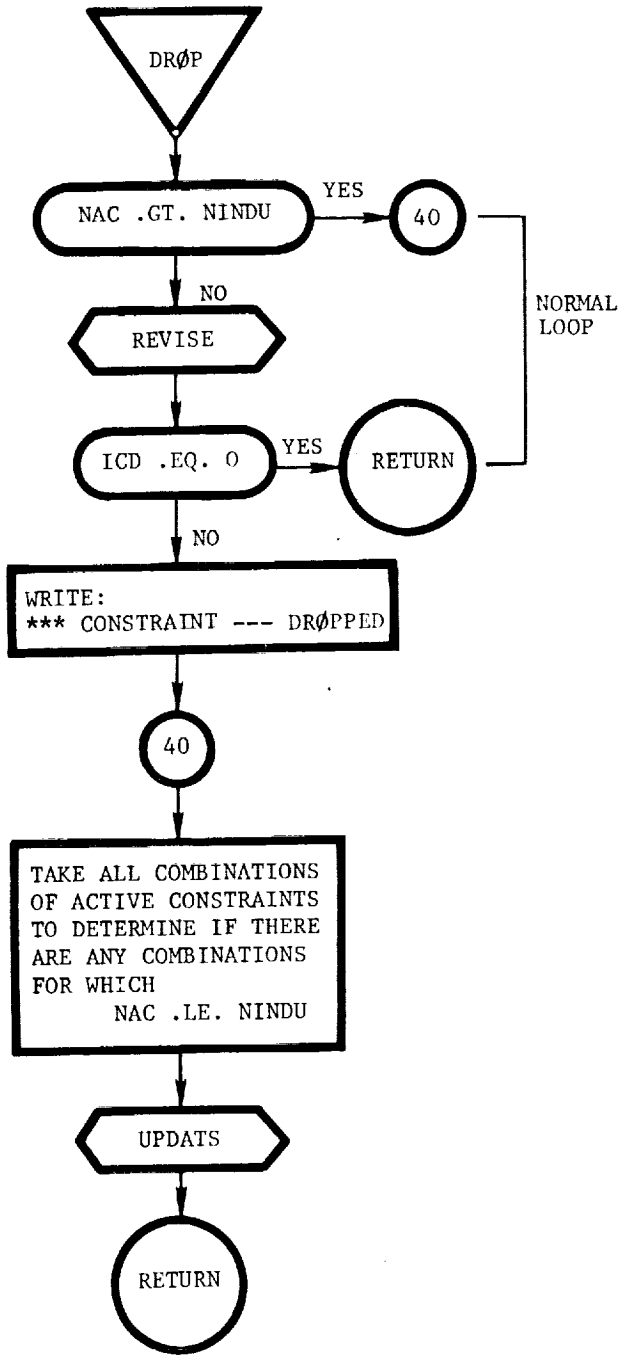
DPRNG: This routine calculates the range based on the dot product of the initial position vector of the vehicle and the impact point vector with the oblate planet.



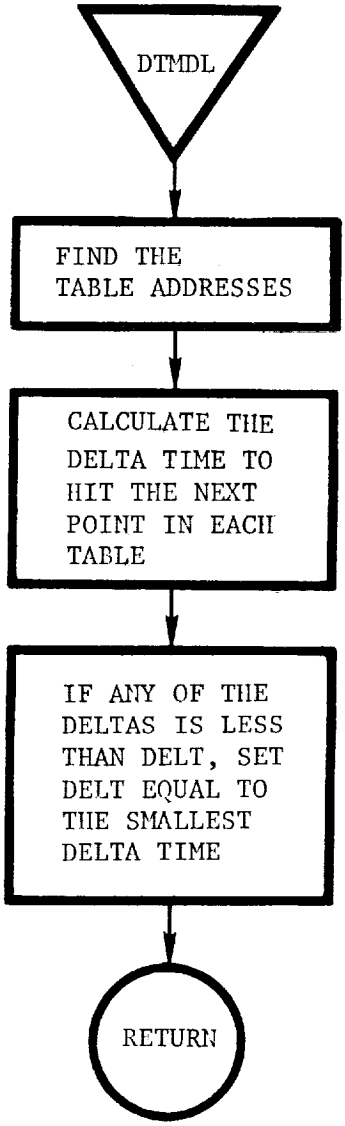
DQUAT: This routine calculates the time derivative of the quaternion parameters.



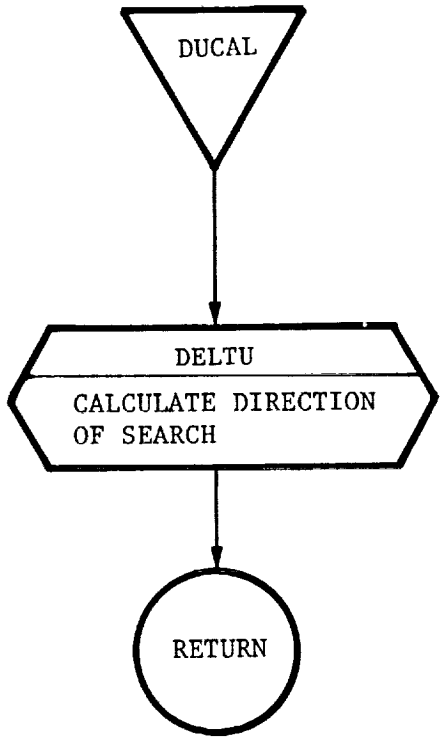
DRØP: This subroutine drops tight constraints.



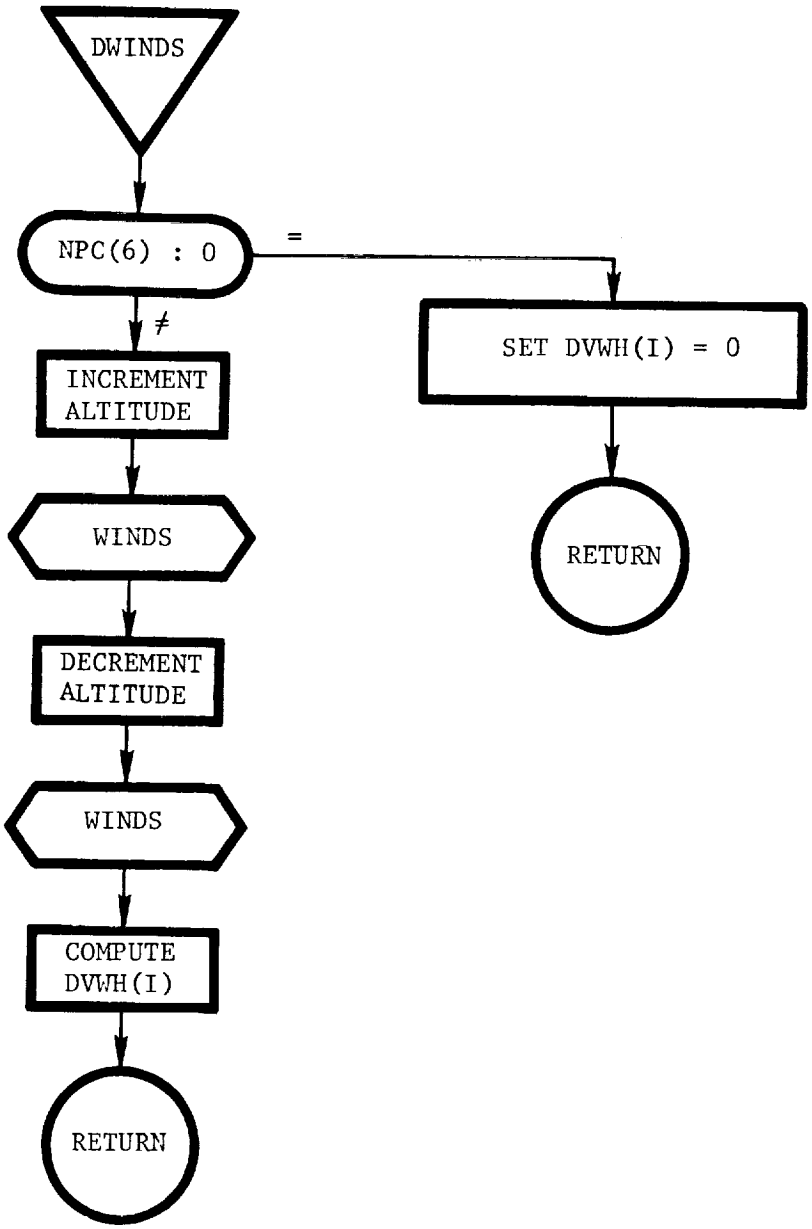
DTMDL: This routine checks the user-specified tables to ensure that the next integration step size is less than or equal to the next time point in any of the tables.



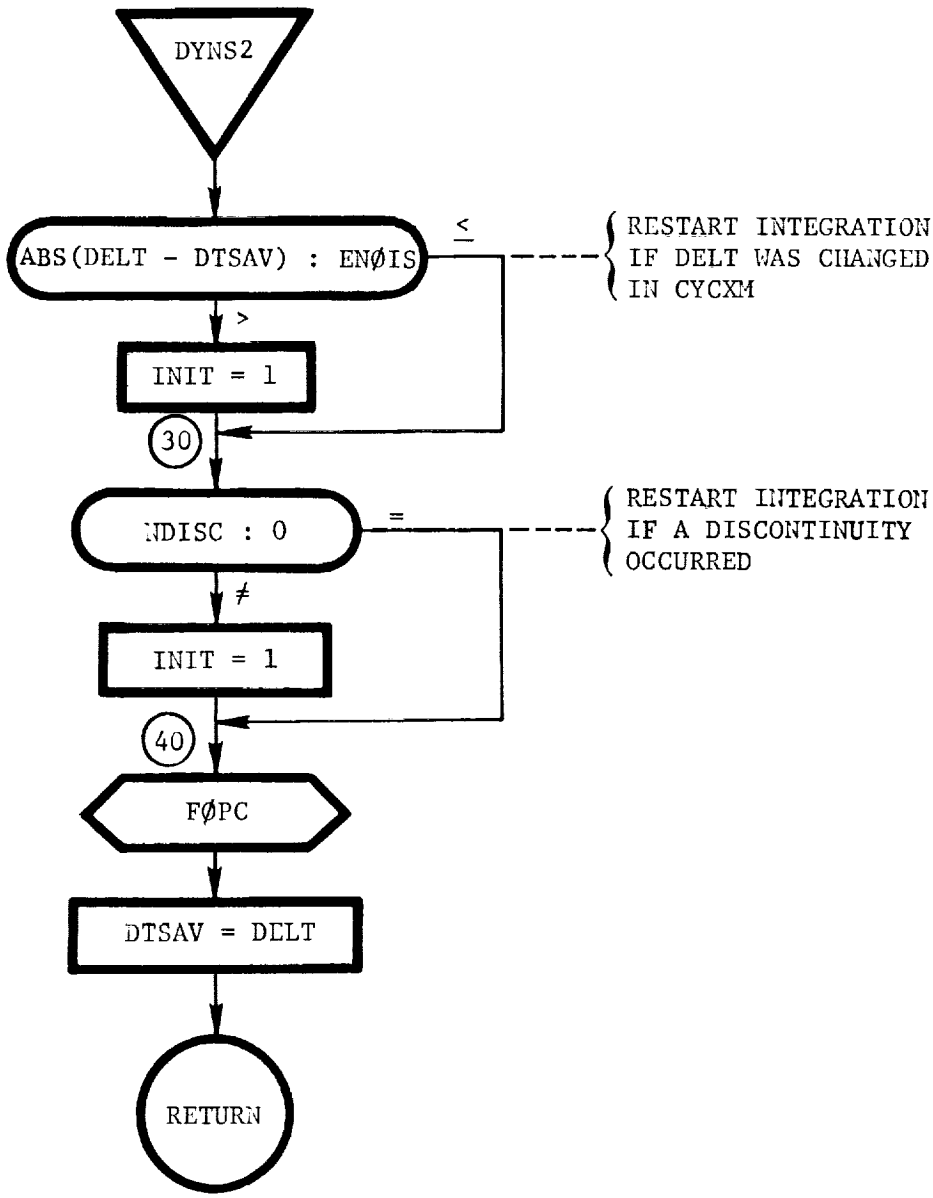
DUCAL: This is the main program of overlay (2.5), and calculates the direction of search by the method specified by SRCHM.



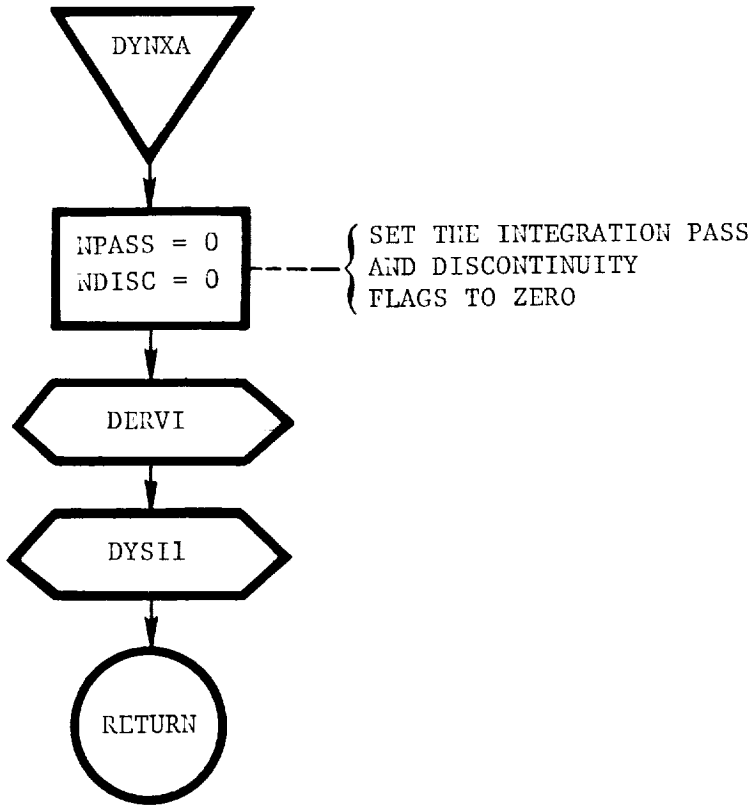
DWINDS: This routine calculates the rate of change in the wind with respect to the altitude above the surface.



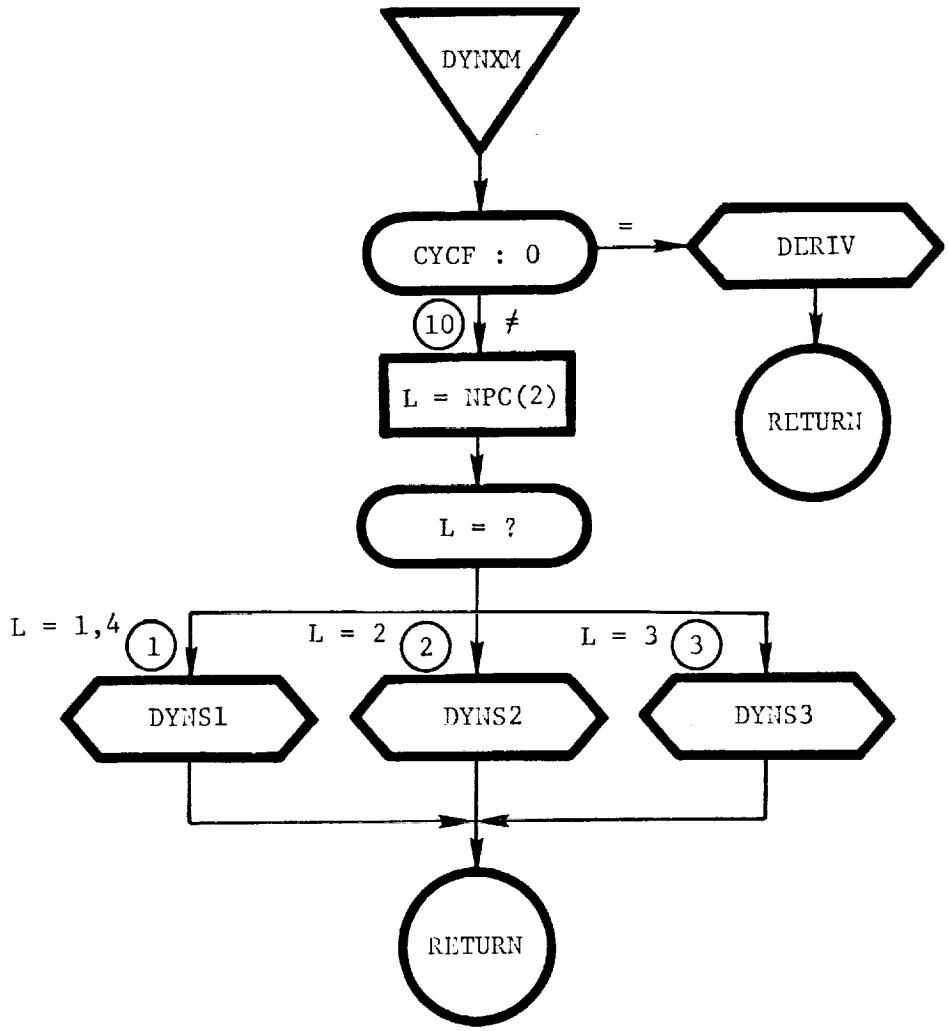
DYNS2: This routine integrates the equations of motion using the predictor-corrector method.



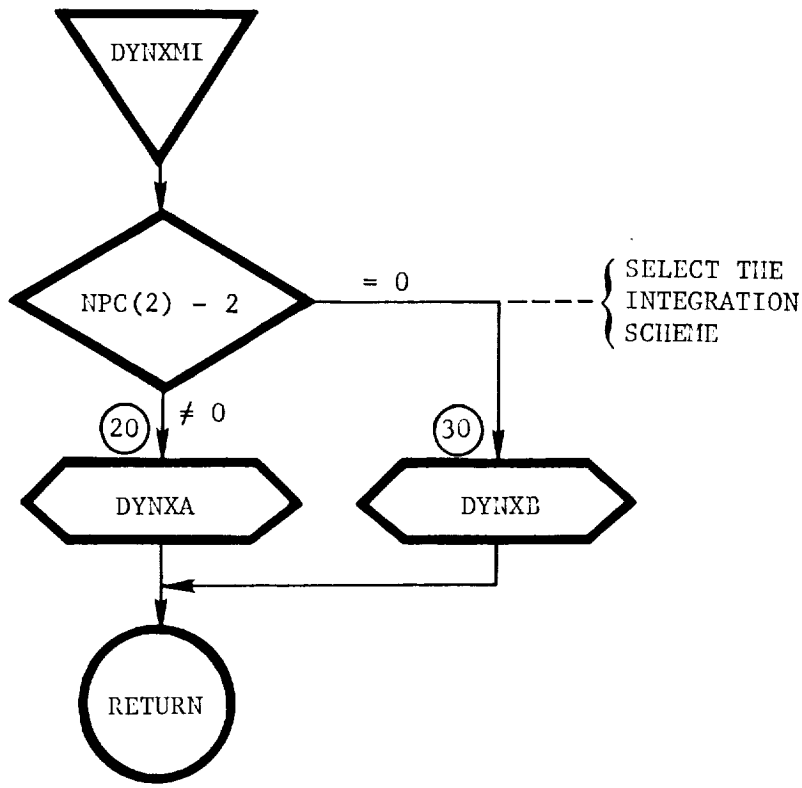
DYNXA: This routine initializes the fourth-order Runge-Kutta integration scheme.



DYNXM: This routine determines which integration scheme is to be used.



DYNXMI: This routine selects the integration scheme to be used.

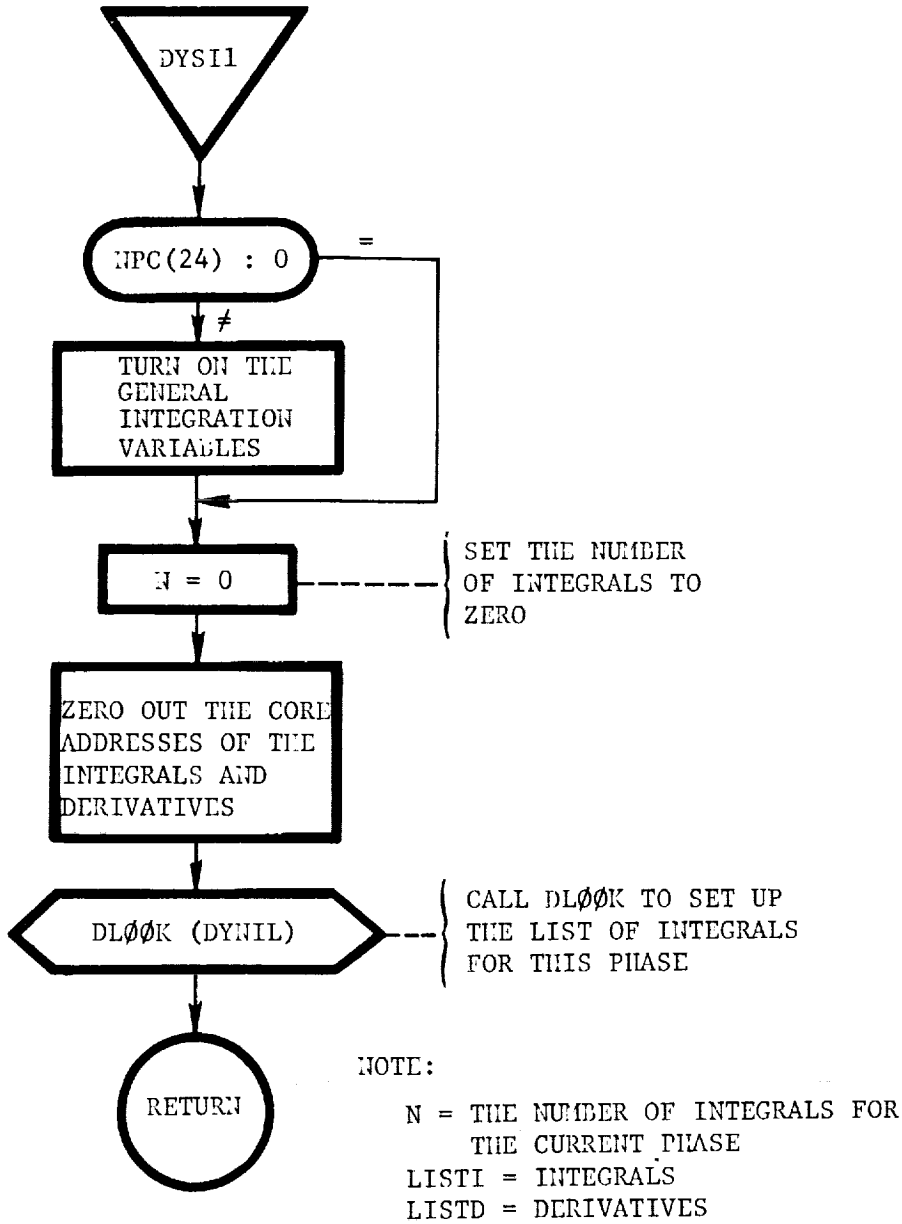


NOTE:

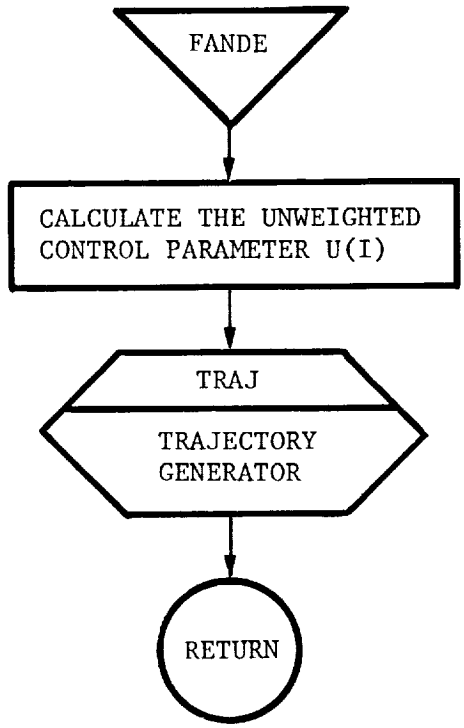
DYNXA - FOURTH-ORDER RUNGE-KUTTA INTEGRATION
- LAPLACE'S METHOD OF INTEGRATION
- ENCKE'S METHOD OF INTEGRATION

DYNXB - FOURTH-ORDER PREDICTOR-CORRECTOR INTEGRATION

DYSI1: This routine sets the integration list.

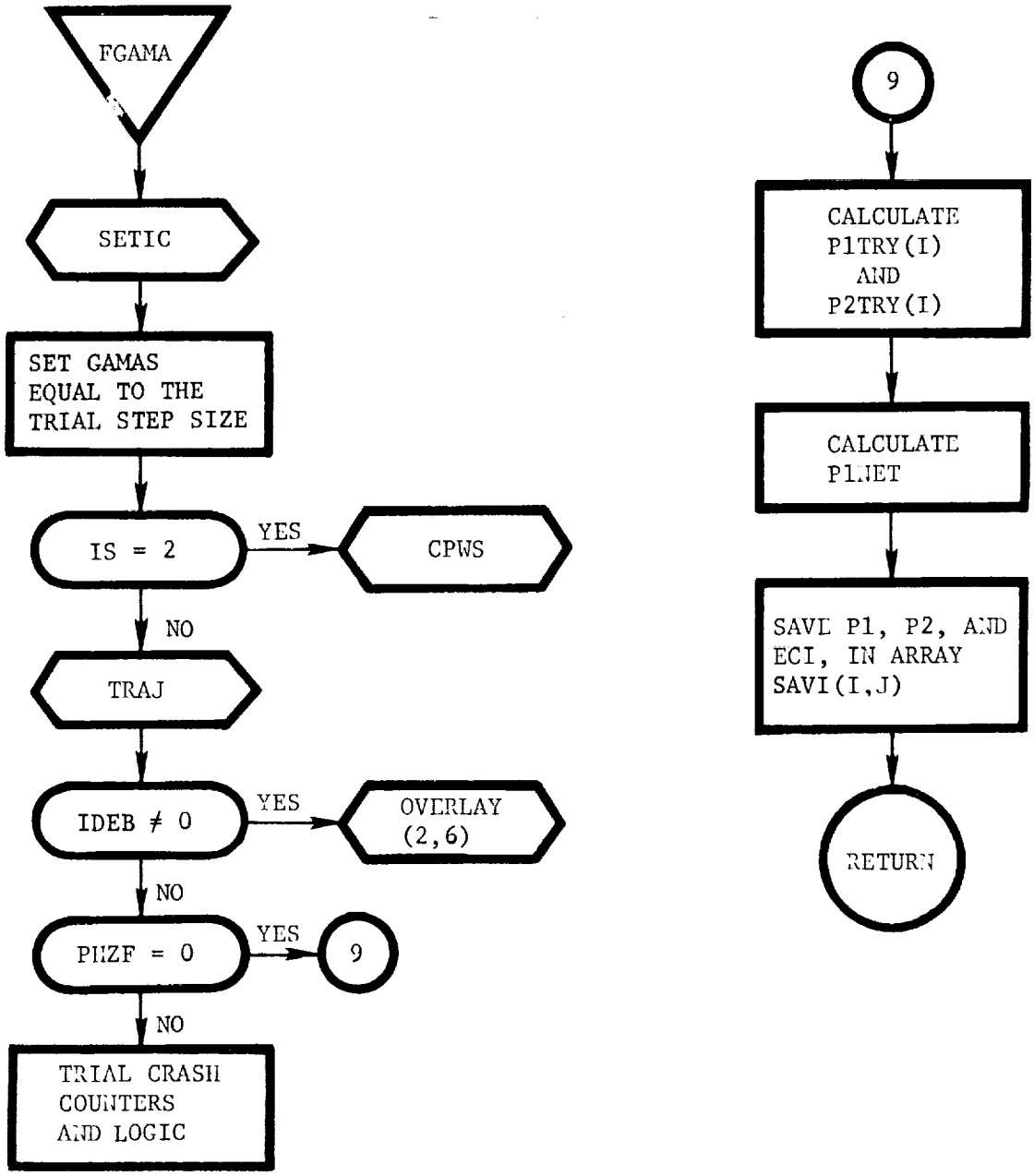


FANDE: This routine computes the unweighted target and optimization variables.

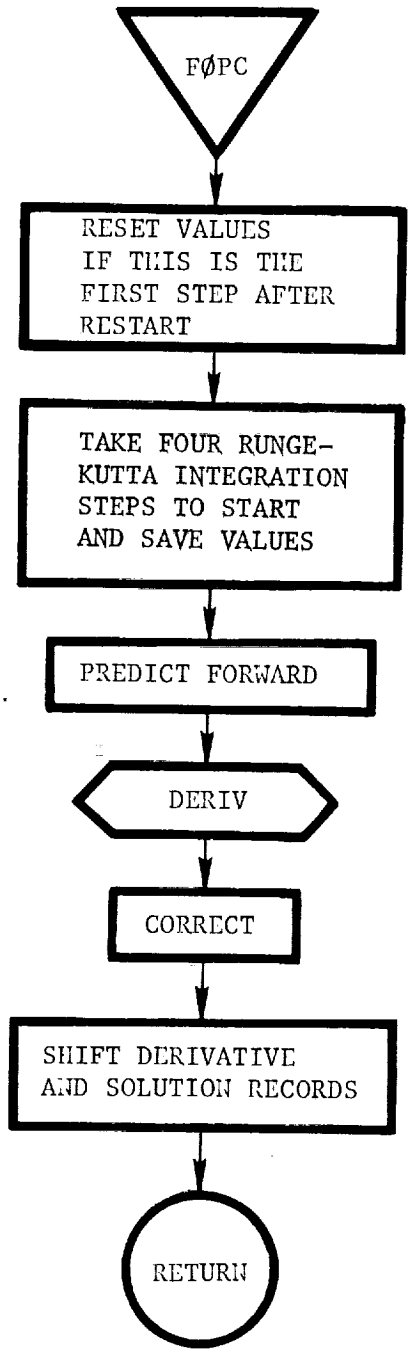


FGAMA(IS): This subroutine calculates the values of P1 and P2 associated with a particular GAMA in the direction of search by changing the controls according to the equation

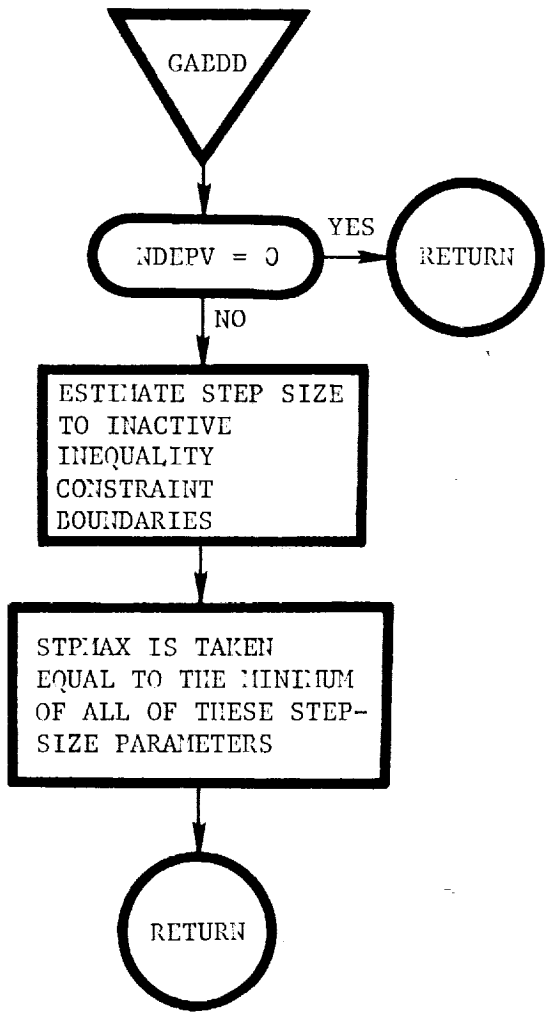
$$U(I) = U(I) + GAMA * DU(I)$$



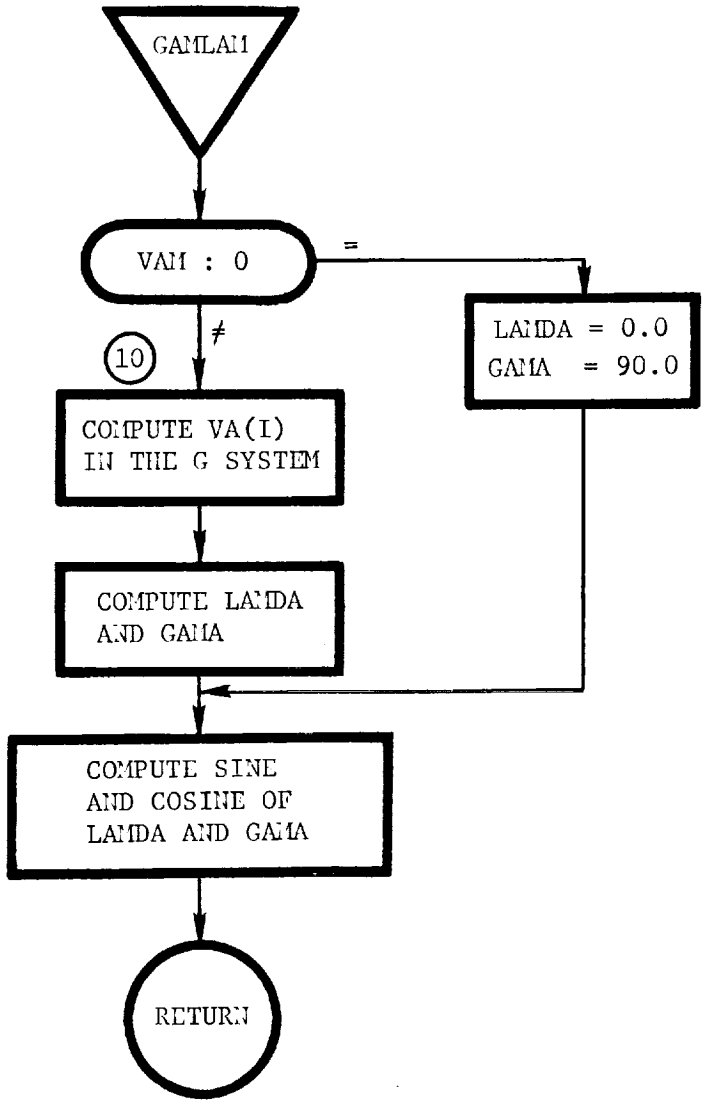
FØPC: This routine contains the fourth-order predictor-corrector integration algorithm.



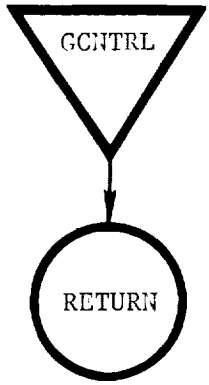
GABDD: This routine computes the maximum step size in the direction of search, STPMAX. STPMAX is computed as the minimum step needed to make one of the inactive inequality constraints become active. This prediction is based on a linearization of the inactive constraints.



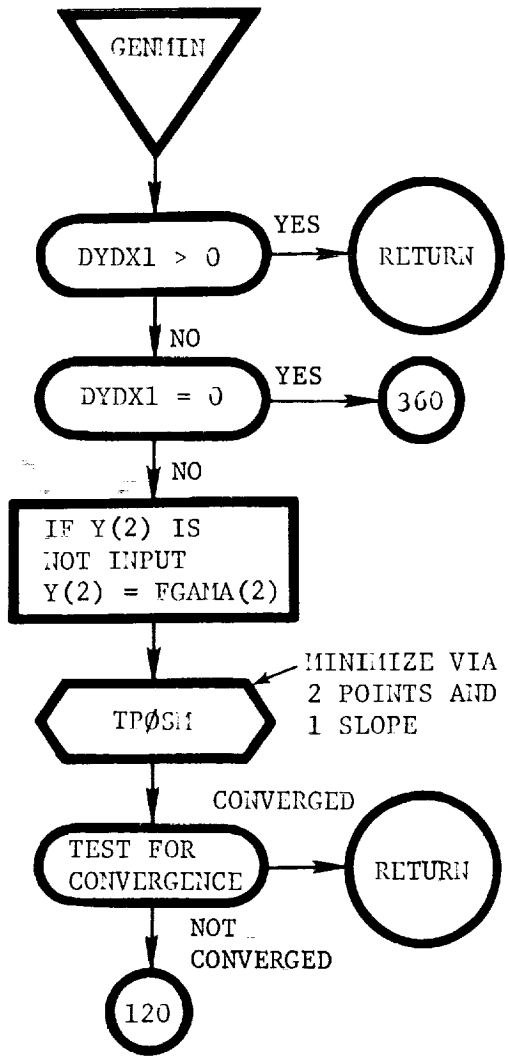
GAMLAM: This routine calculates the flight path angle and azimuth angle of the atmospheric relative velocity vector.

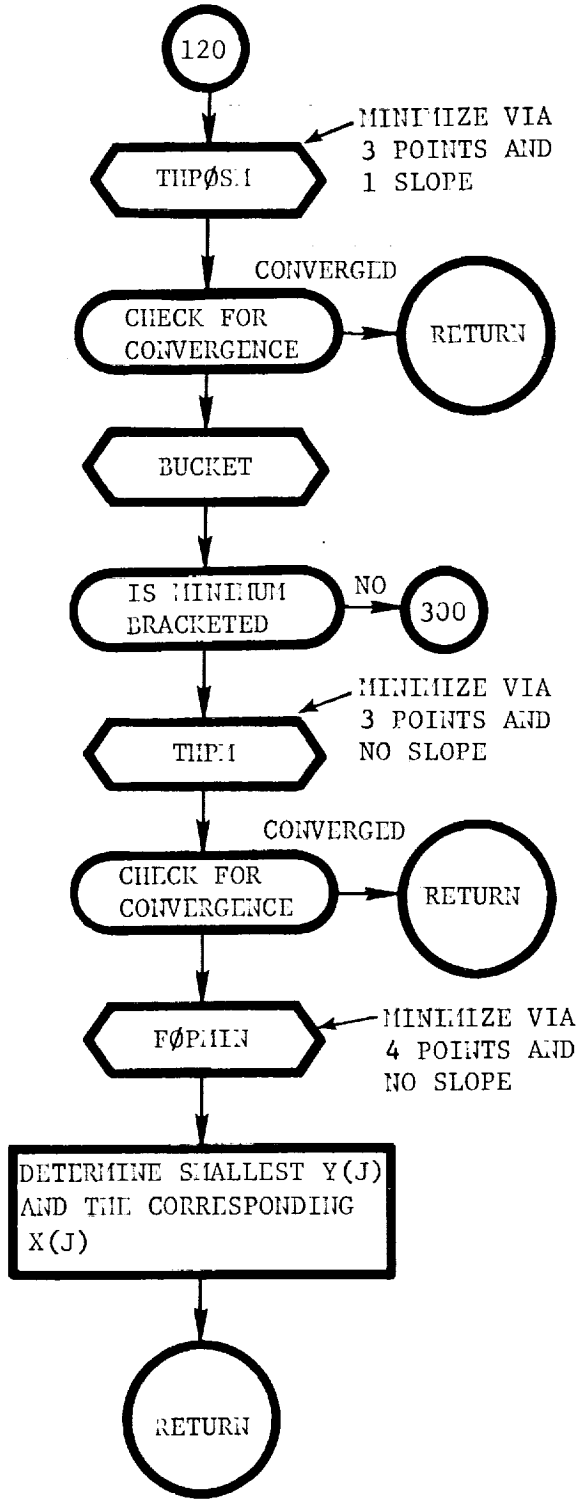


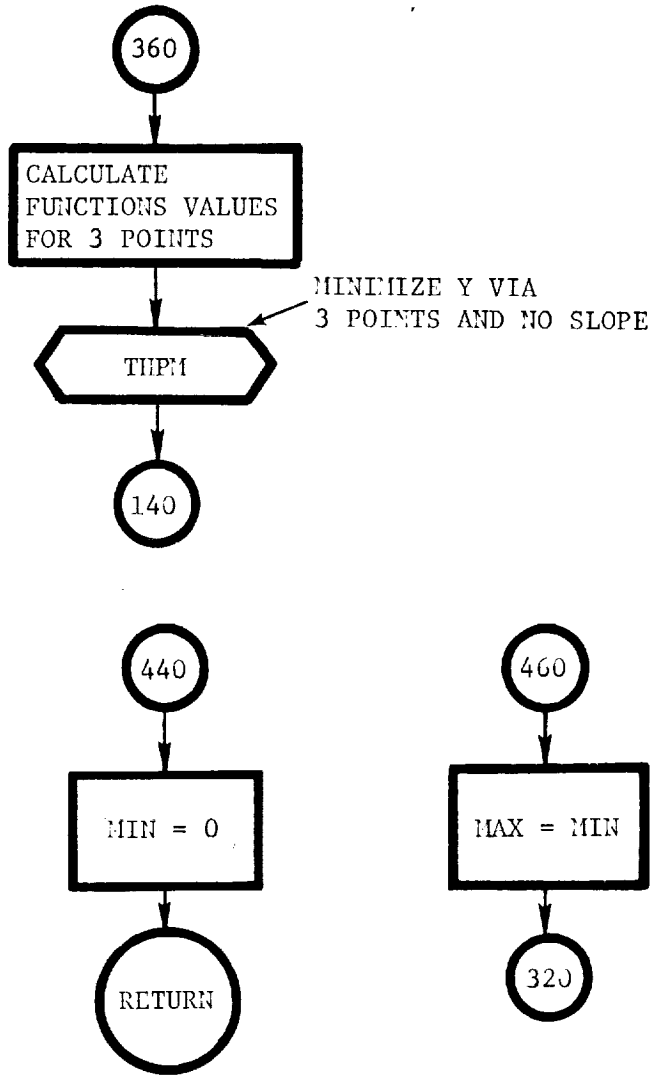
GCNTRL: This is a blank routine used for simulating the hardware lags and errors associated with implementing the closed-loop guidance steering commands.



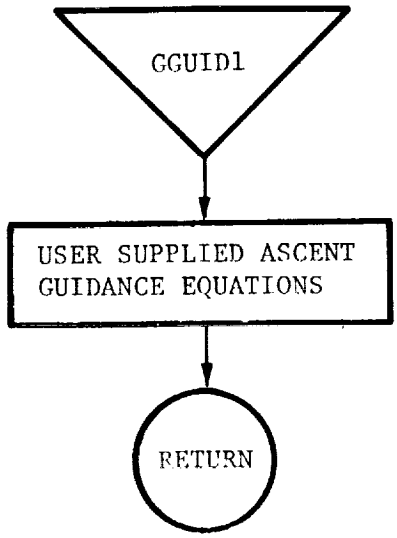
GENMIN: This routine finds the minimum of the function y, given the pairs (X, Y) and the slope at X = 0. If the pairs (X, Y) are not given, then GENMIN generates these pairs automatically. The minimization is based on approximations to the functions that are made using quadratic and cubic polynomials. The analytically calculated minimums of these polynomials are used as estimates to the actual minimum values of y.



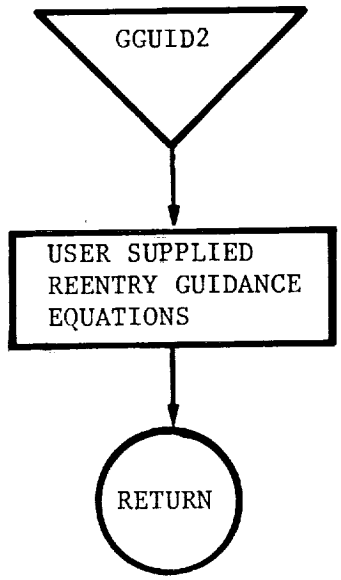

120
THPØS.1
MINIMIZE VIA
3 POINTS AND
1 SLOPE
CONVERGED
CHECK FOR
CONVERGENCE
RETURN
BUCKET
IS MINIMUM
BRACKETED
NO
300
THP.1
MINIMIZE VIA
3 POINTS AND
NO SLOPE
CONVERGED
CHECK FOR
CONVERGENCE
RETURN
FØPMIN
MINIMIZE VIA
4 POINTS AND
NO SLOPE
DETERMINE SMALLEST Y(J)
AND THE CORRESPONDING
X(J)
RETURN


360
CALCULATE
FUNCTIONS VALUES
FOR 3 POINTS
MINIMIZE Y VIA
3 POINTS AND NO SLOPE
THPM
140
440
MIN = 0
RETURN
460
MAX = MIN
320

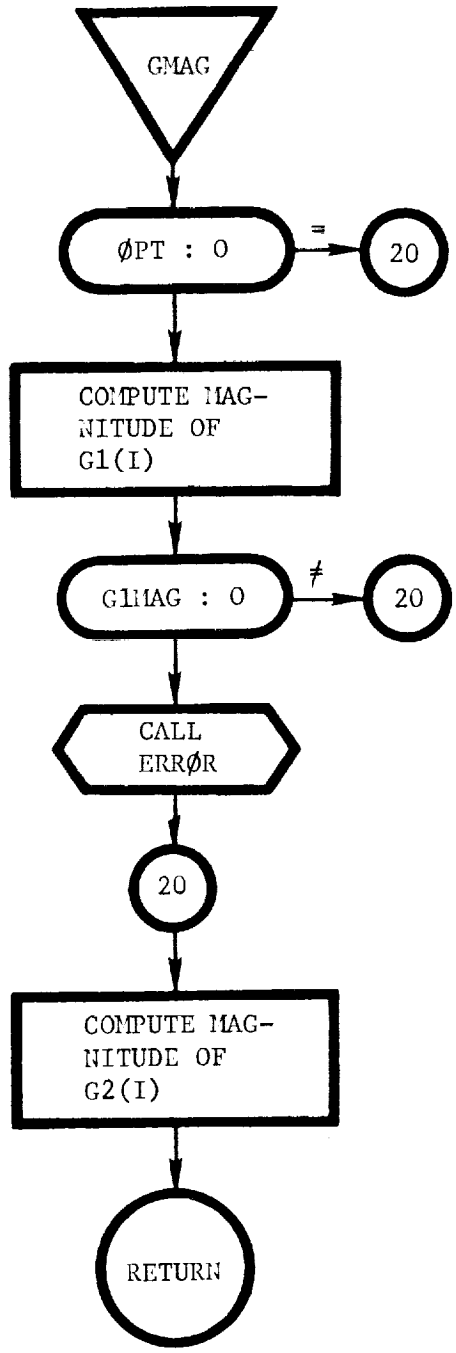
GGUID1: This routine contains user supplied ascent guidance equations.



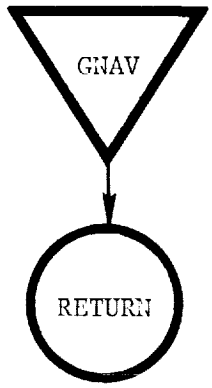
GGUID2: This routine contains user supplied reentry guidance equations.



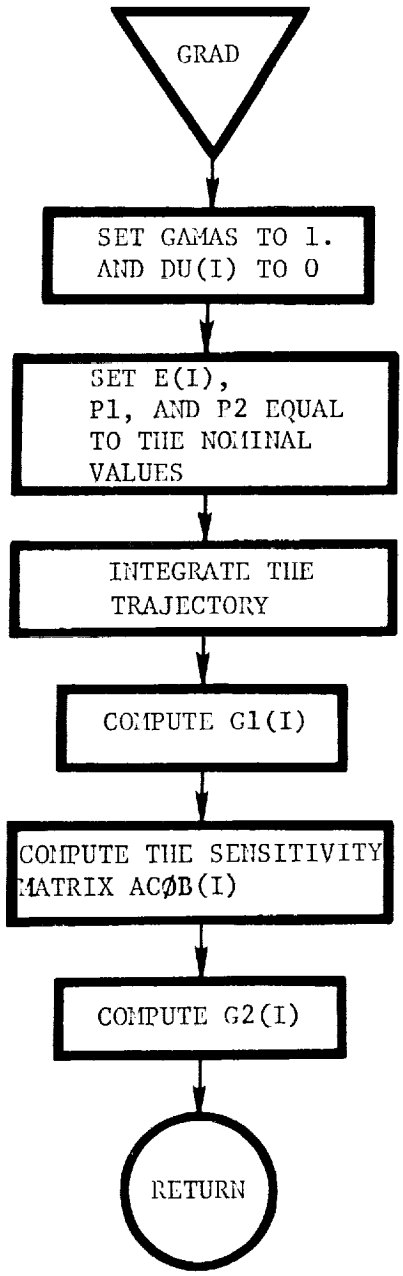
GMAG: This routine computes the magnitude of the gradient vectors G1(I) and G2(I).



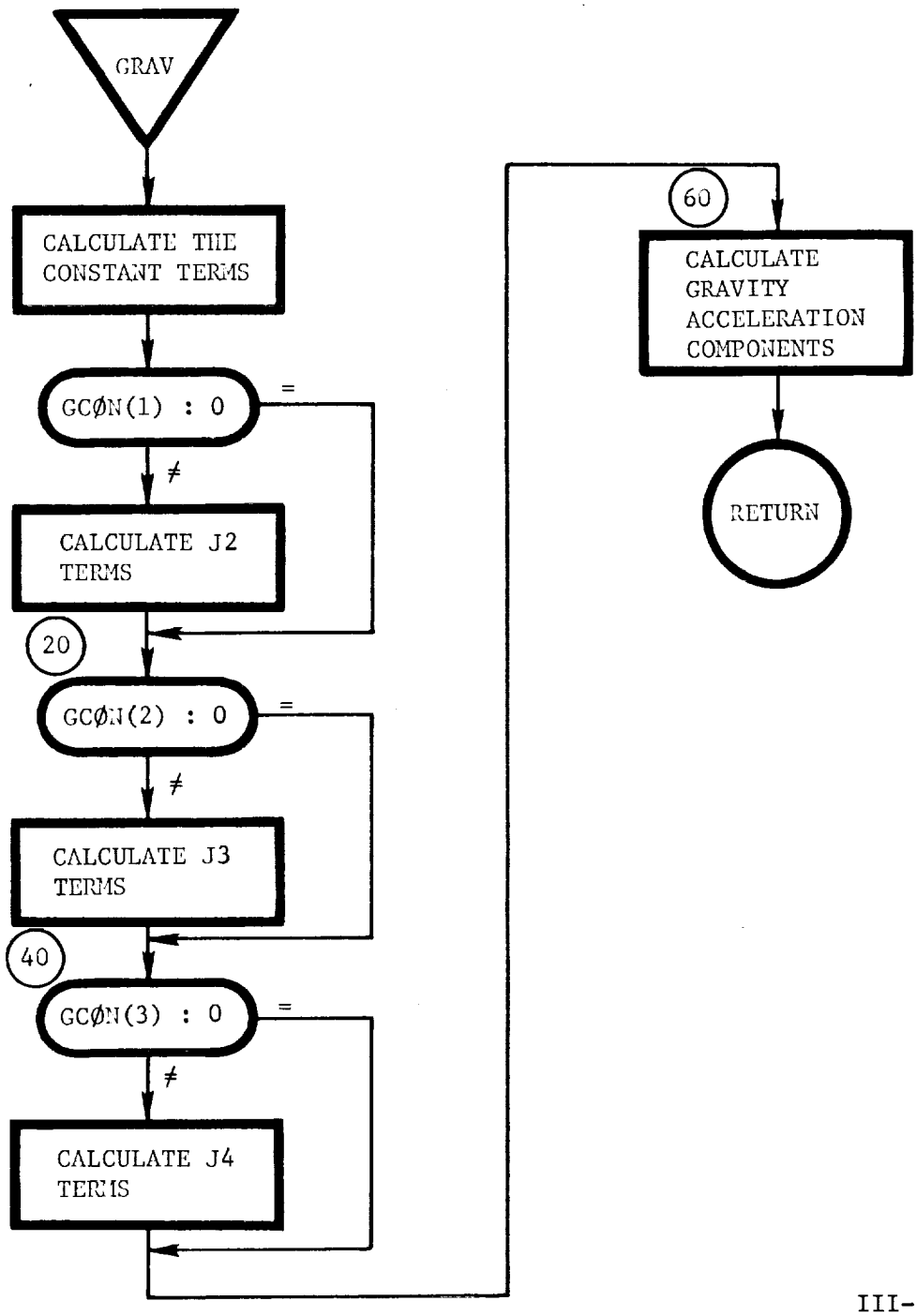
GNAV: This is a blank routine that is to be used for simulating the navigation equations for the closed-loop guidance system being analyzed.



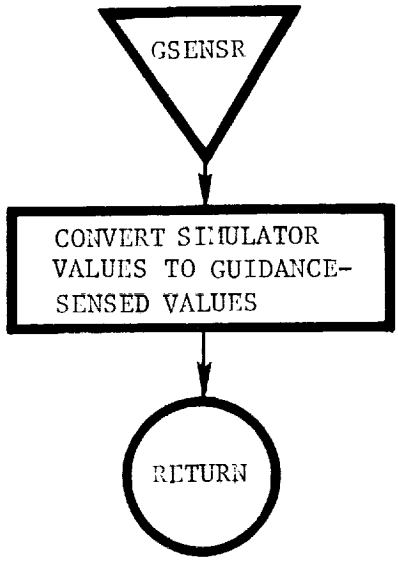
GRAD: This routine computes the gradients of the penalty functions (G1 and G2) with respect to the control parameters.



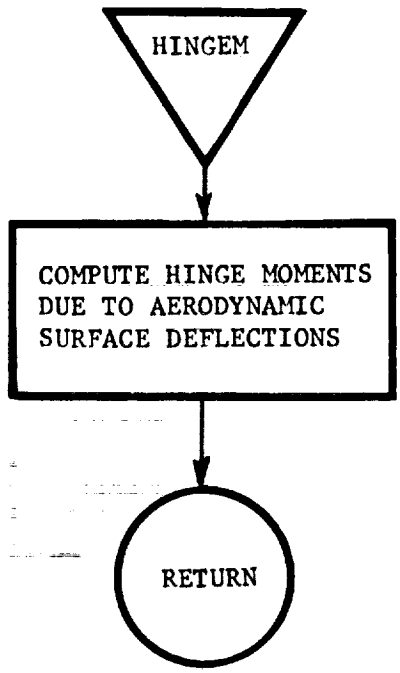
GRAV: This routine calculates the gravity acceleration vector.



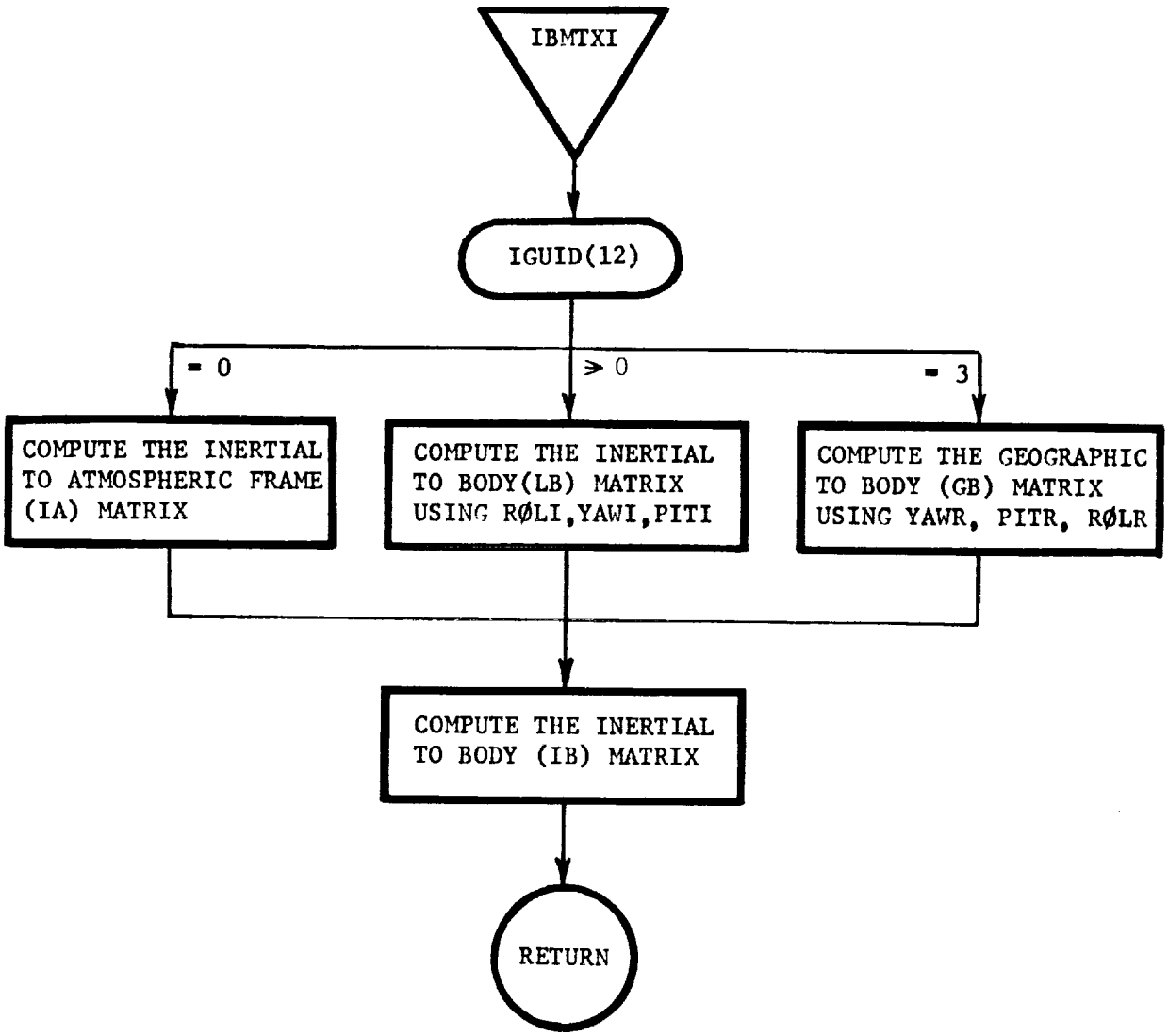
GSENSR: This routine is to be used for simulating the interface between the trajectory simulator (real world) and the guidance sensor (hardware detected) being analyzed.



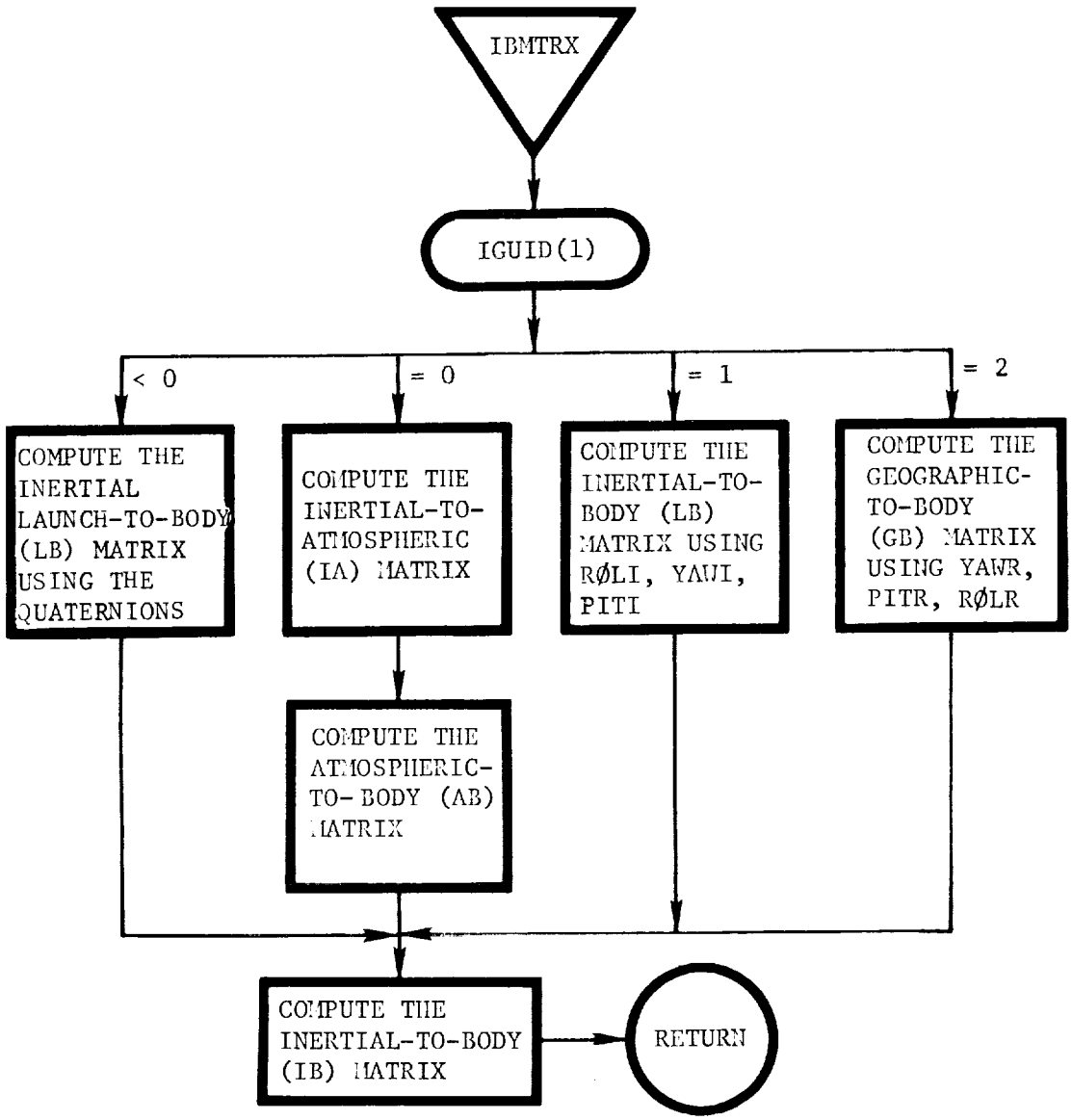
HINGEM: This routine calculates the aerodynamic hinge moments due to flap deflections.



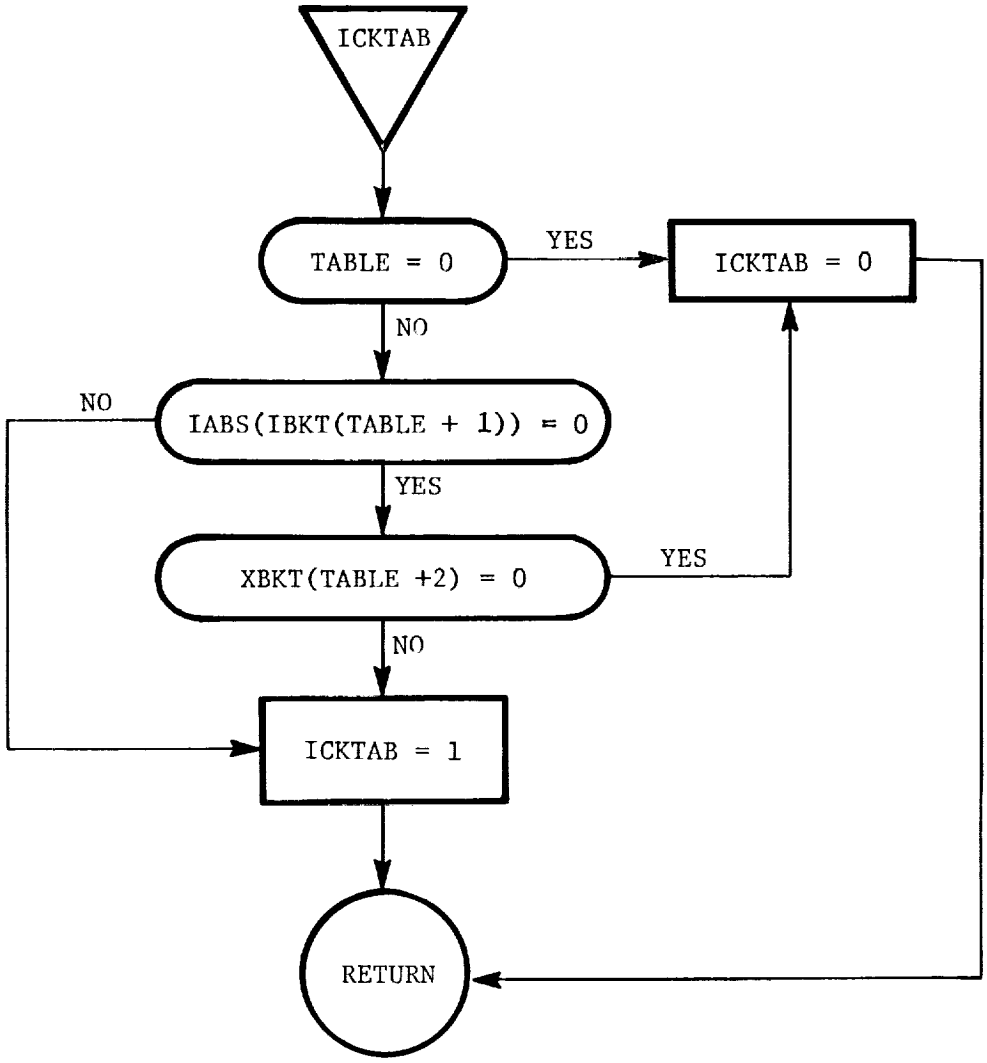
IBMTXI: This routine calculates the IB matrix from aerodynamic or Euler angles. The IB matrix is used to initialize the quaternion rate equations.



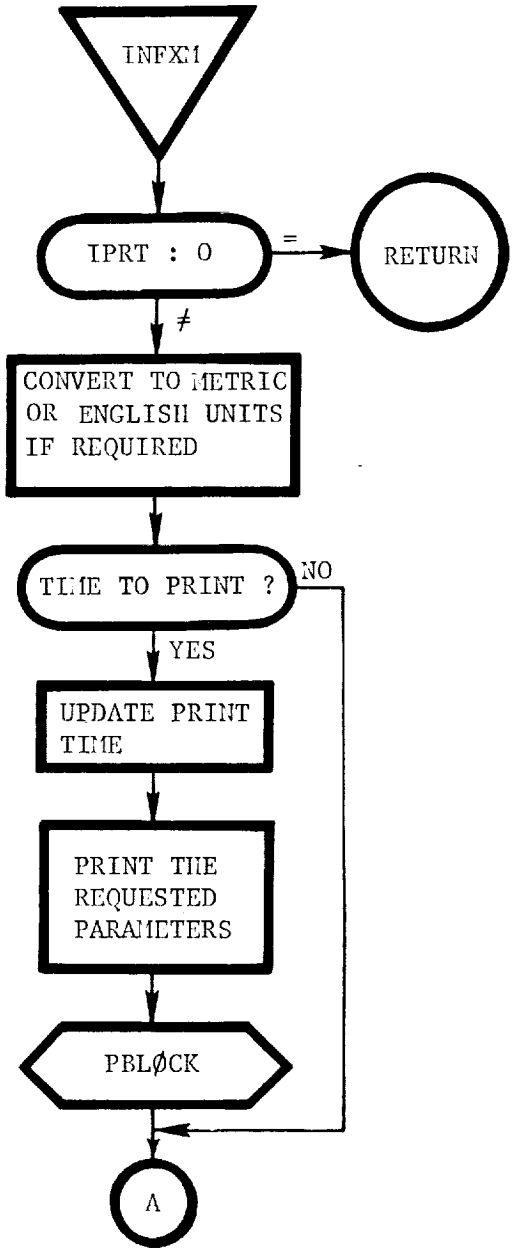
IBMTRX: This routine calculates the inertial-to-body (IB) matrix, based on the guidance (steering) option selected by user input.

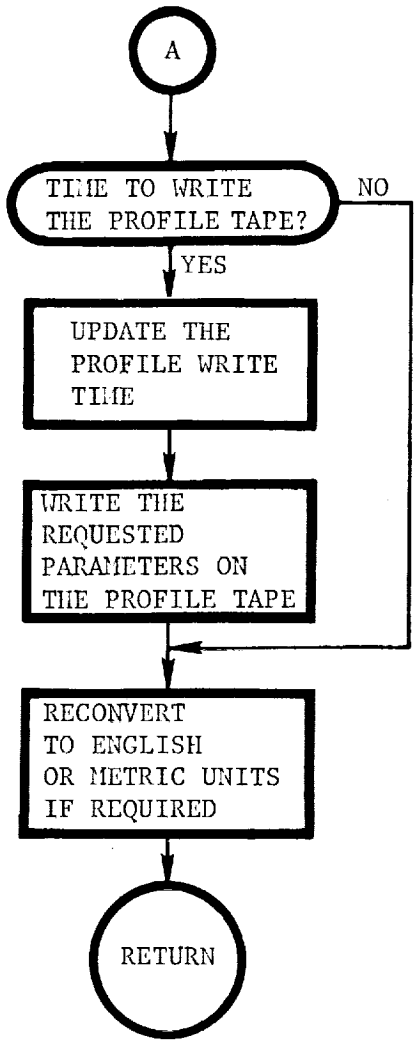


ICKTAB: This function serves as a flag indicating whether a table was input, and if input, whether it is a constant table of value 0.

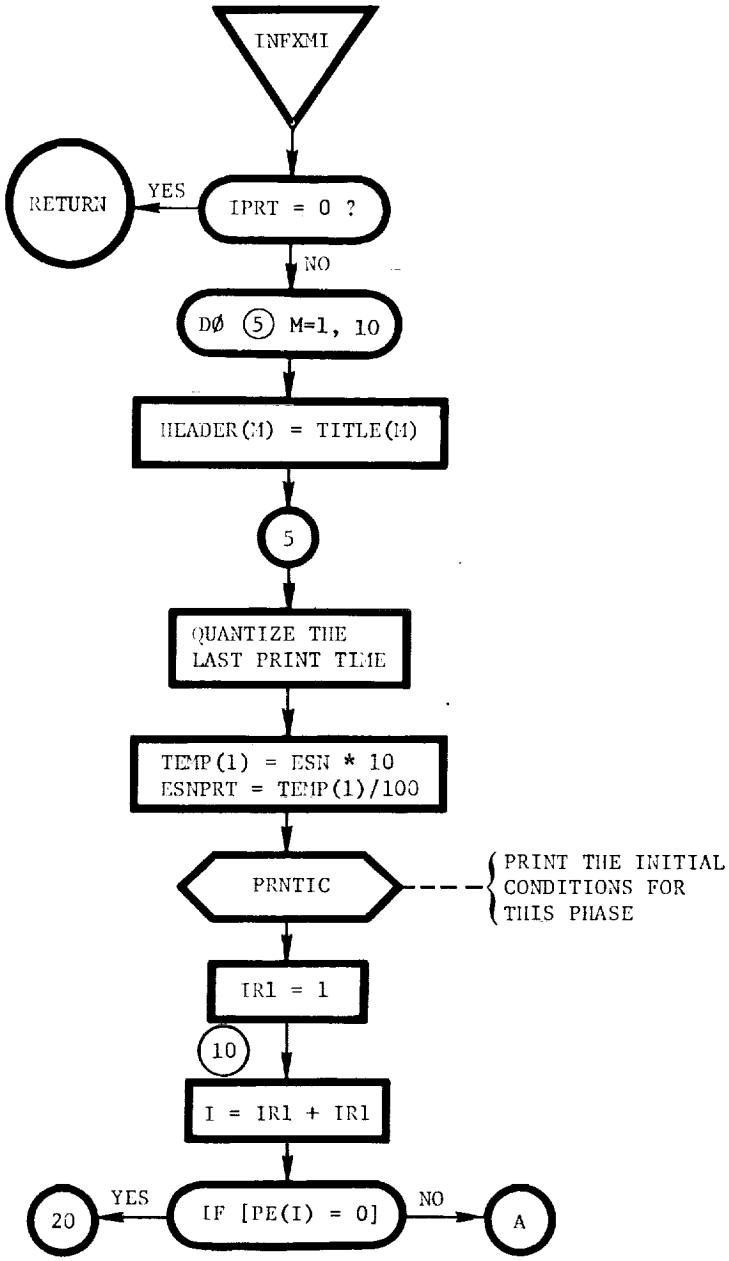


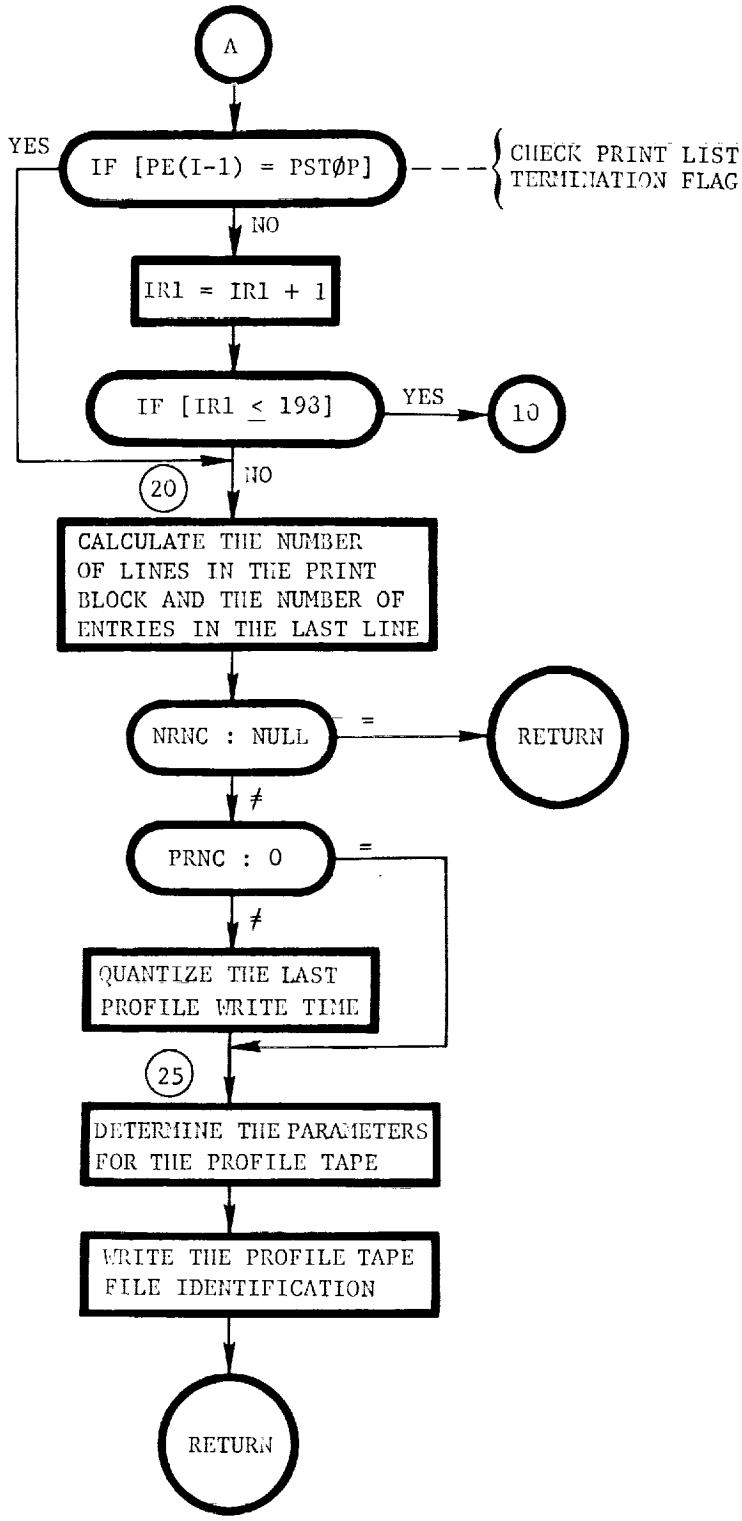
INFXM: This routine performs the output data processing functions. It also calls subroutine CØNIC, depending on the conic calculation option requested by NPC(1).



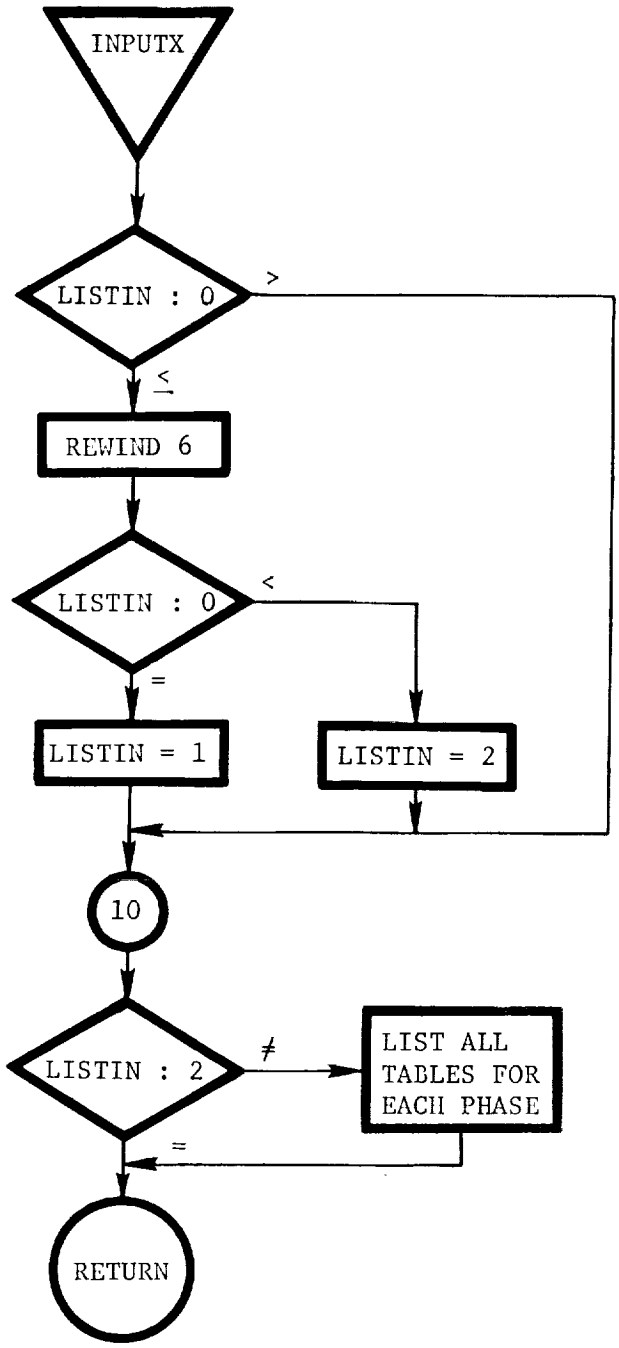

A
TIME TO WRITE THE PROFILE TAPE?
NO
YES
UPDATE THE PROFILE WRITE TIME
WRITE THE REQUESTED PARAMETERS ON THE PROFILE TAPE
RECONVERT TO ENGLISH OR METRIC UNITS IF REQUIRED
RETURN

INFXMI: This routine determines which variables are to be printed and which variables are to be written on the profile tape.




A
YES
IF [PE(I-1) = PSTØP]
CHECK PRINT LIST TERMINATION FLAG
NO
IR1 = IR1 + 1
IF [IR1 ≤ 193]
YES
10
20
NO
CALCULATE THE NUMBER OF LINES IN THE PRINT BLOCK AND THE NUMBER OF ENTRIES IN THE LAST LINE
NRNC : NULL
=
RETURN
≠
PRNC : 0
=
≠
QUANTIZE THE LAST PROFILE WRITE TIME
25
DETERMINE THE PARAMETERS FOR THE PROFILE TAPE
WRITE THE PROFILE TAPE FILE IDENTIFICATION
RETURN

INPUTX: This routine prints a summary of the input table data. The variable LISTIN is also checked to determine whether or not to rewind the output file, which eliminates the listing of the input data deck.



INTGRL (LIST, NUM, KEY): This routine initializes the list of variables to be integrated. The list of variables to be integrated is called DYNIL. It contains 148 cells that are stored in subroutine BLKDAT. There are three cells for each integrated variable, which corresponds to a total of 49 integration variables. The first cell indicates the total size of the array, the second cell contains the Hollerith name of the first variable to be integrated, the third cell contains the Hollerith name of its derivative, and the fourth cell contains a flag to indicate whether or not to integrate that variable, etc. If the flag is zero, the variable is not integrated; if the flag is equal to 1, the variable is integrated.

INTGRL is also used to turn the integration of variables on or off as desired. For example, if NPC(11) = 1, we wish to activate the inequality constraint integrations (i.e., the integration of FVAL1, FVAL2, FVAL3). In this case, subroutine MØTIAL calls INTGRL as follows to activate the integration:

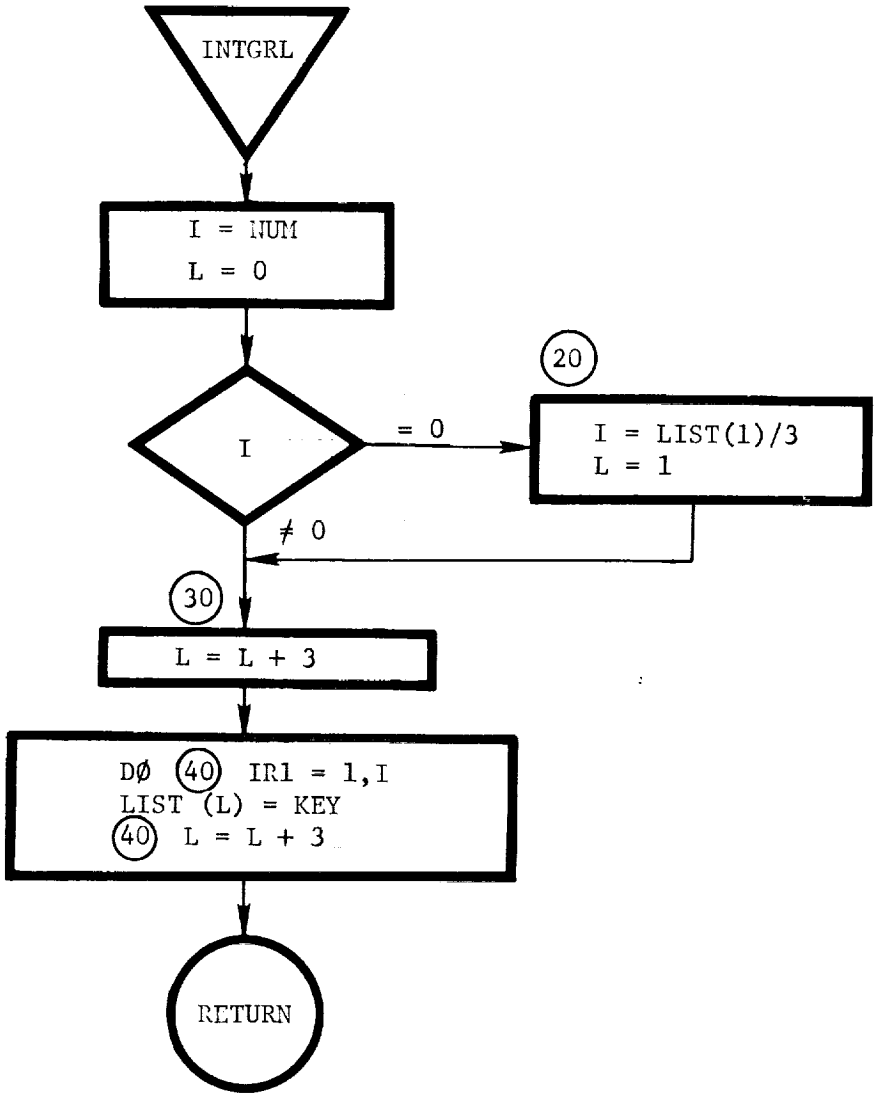
```
CALL INTGRL (DYNIL(38), N03, N01)
```

where:

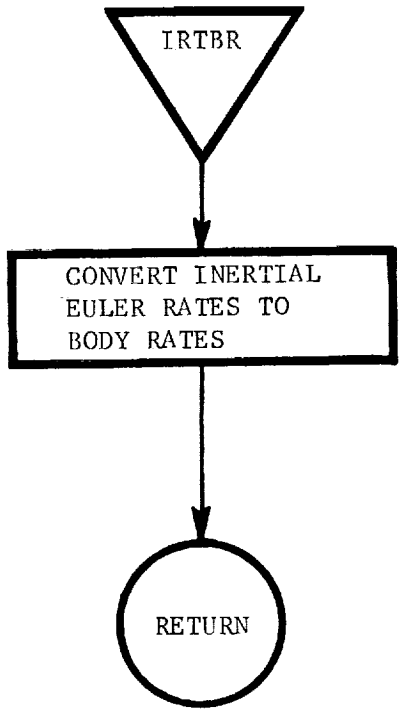
DYNIL(38) = the position of FVAL1 in the array DYNIL

N03 = fixed point 3, which means that the three integrals namely, FVAL1, FVAL2, and FVAL3, are to be turned on, since they are in sequence.

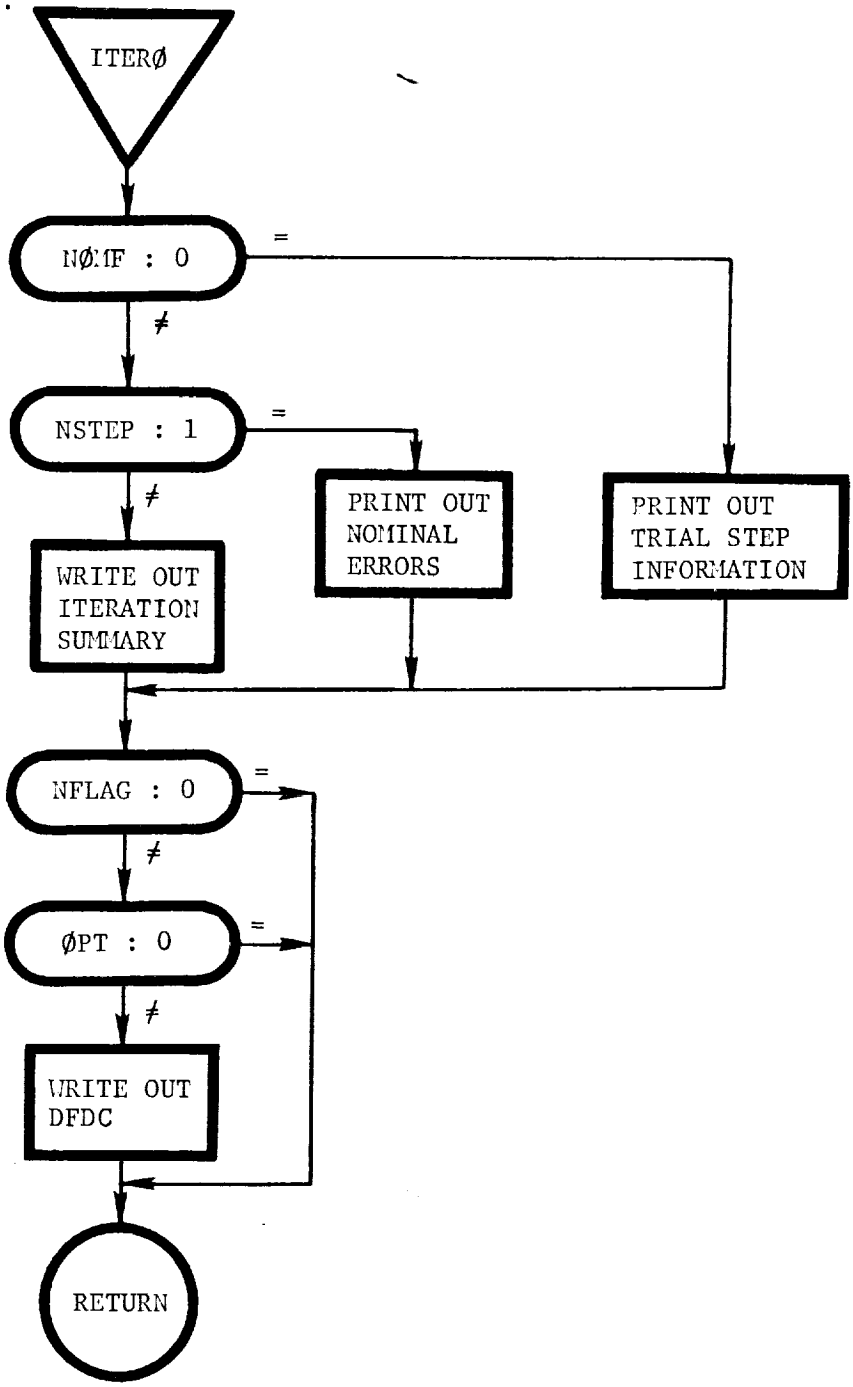
N01 = fixed point 1; this means turn on the integration of the variables. If the argument were N00 (fixed-point zero), the integration would be turned off.


INTGRL
I = NUM
L = 0
I
= 0
20
I = LIST(1)/3
L = 1
≠ 0
30
L = L + 3
DØ 40 IR1 = 1,I
LIST (L) = KEY
40 L = L + 3
RETURN

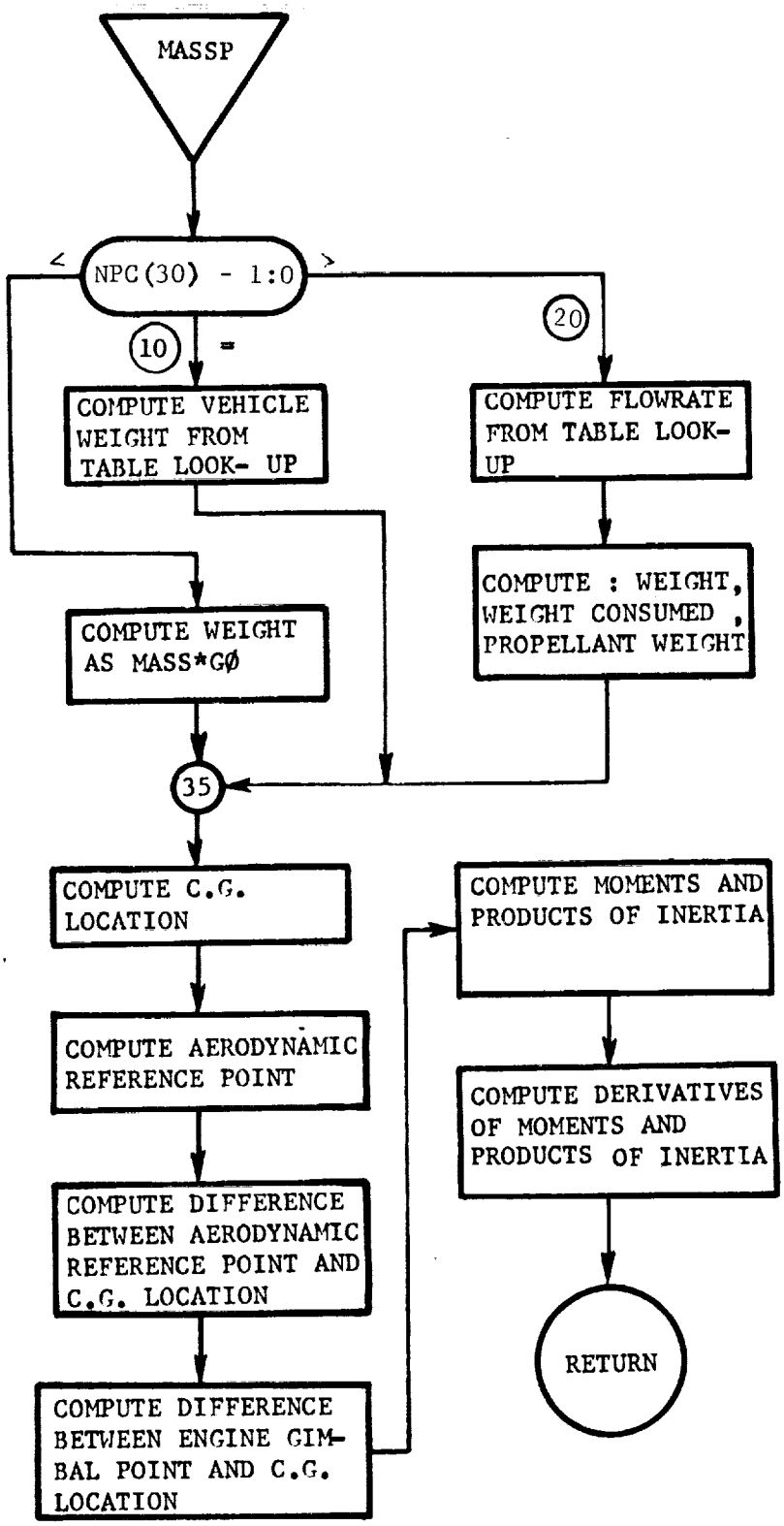
IRTBR: This routine calculates body rates from inertial Euler angle rates.



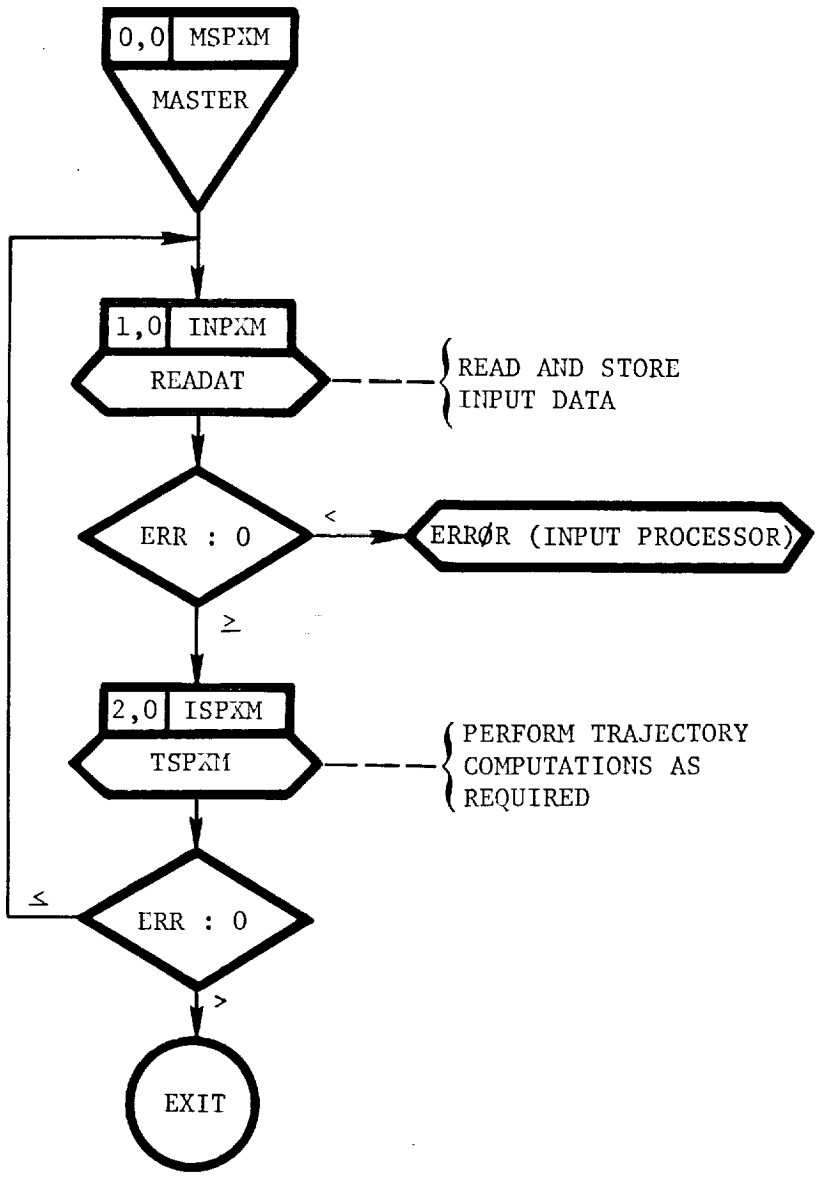
ITERØ: Main program of overlay (2,6). This routine prints out the iteration summary as required. The iteration summary contains all the information relating to the search/optimization algorithm.



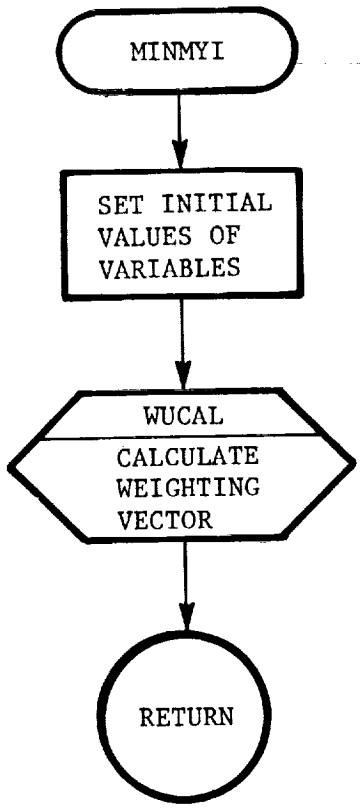
MASSP: This routine calculates the mass properties of the vehicle.



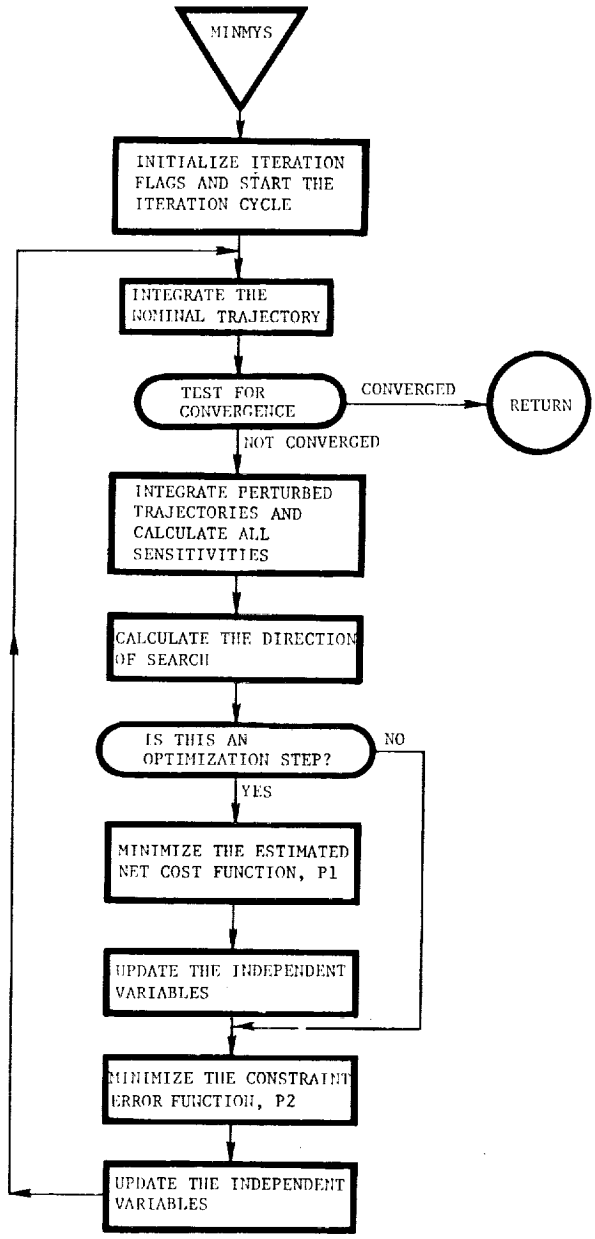
MASTER: This is the main program of overlay (0,0). It decides whether to read input data or execute the problem.



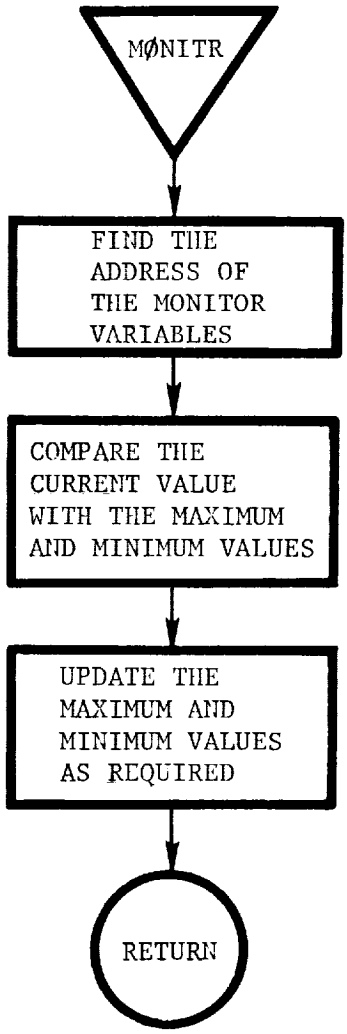
MINMYI: This routine initializes the targeting and optimization parameters.



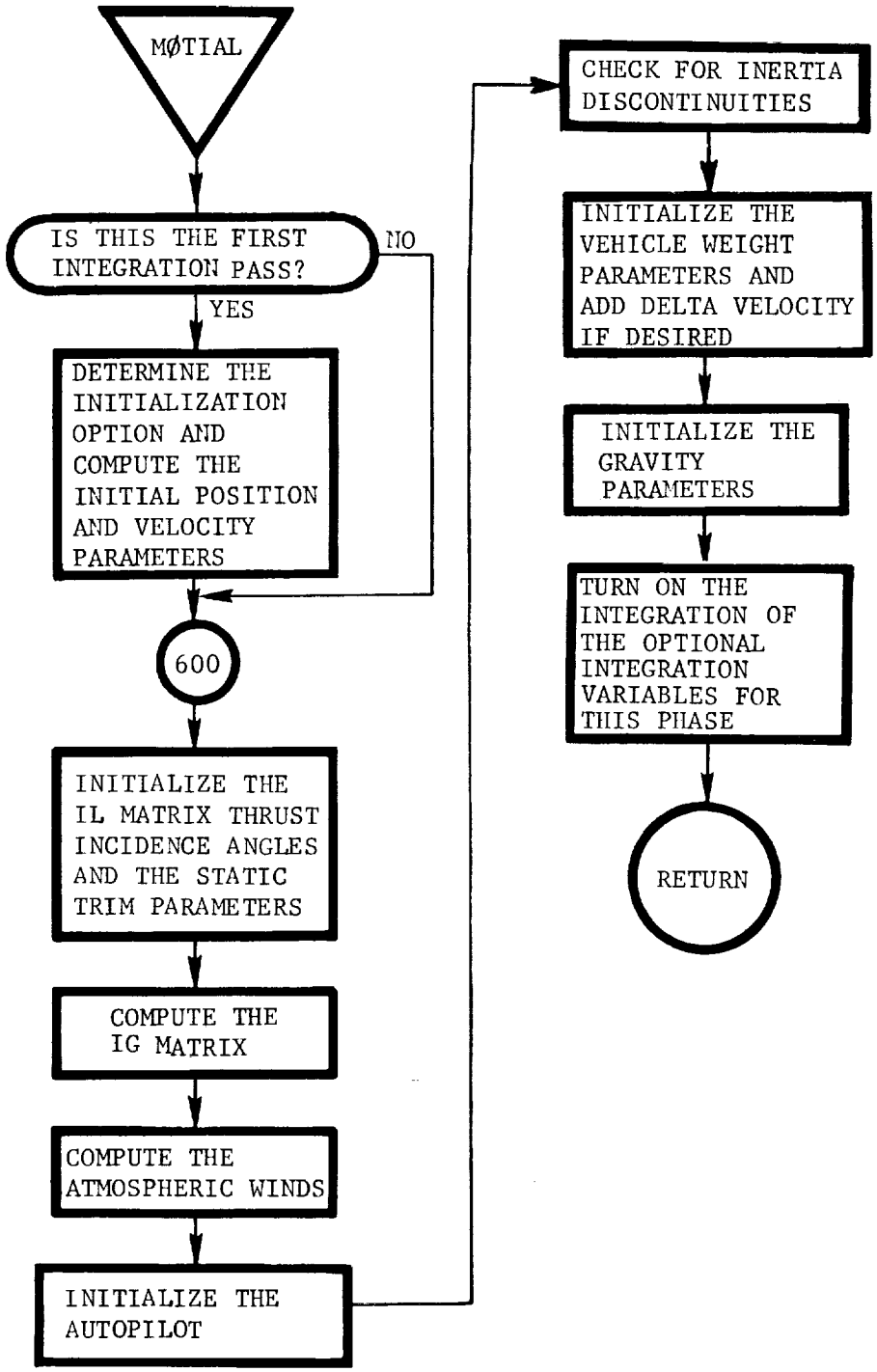
MINMYS: This routine contains the optimization executive logic for the various options that are available. It phases the various iteration paths by examining various properties of the objective function and the constraint manifolds.



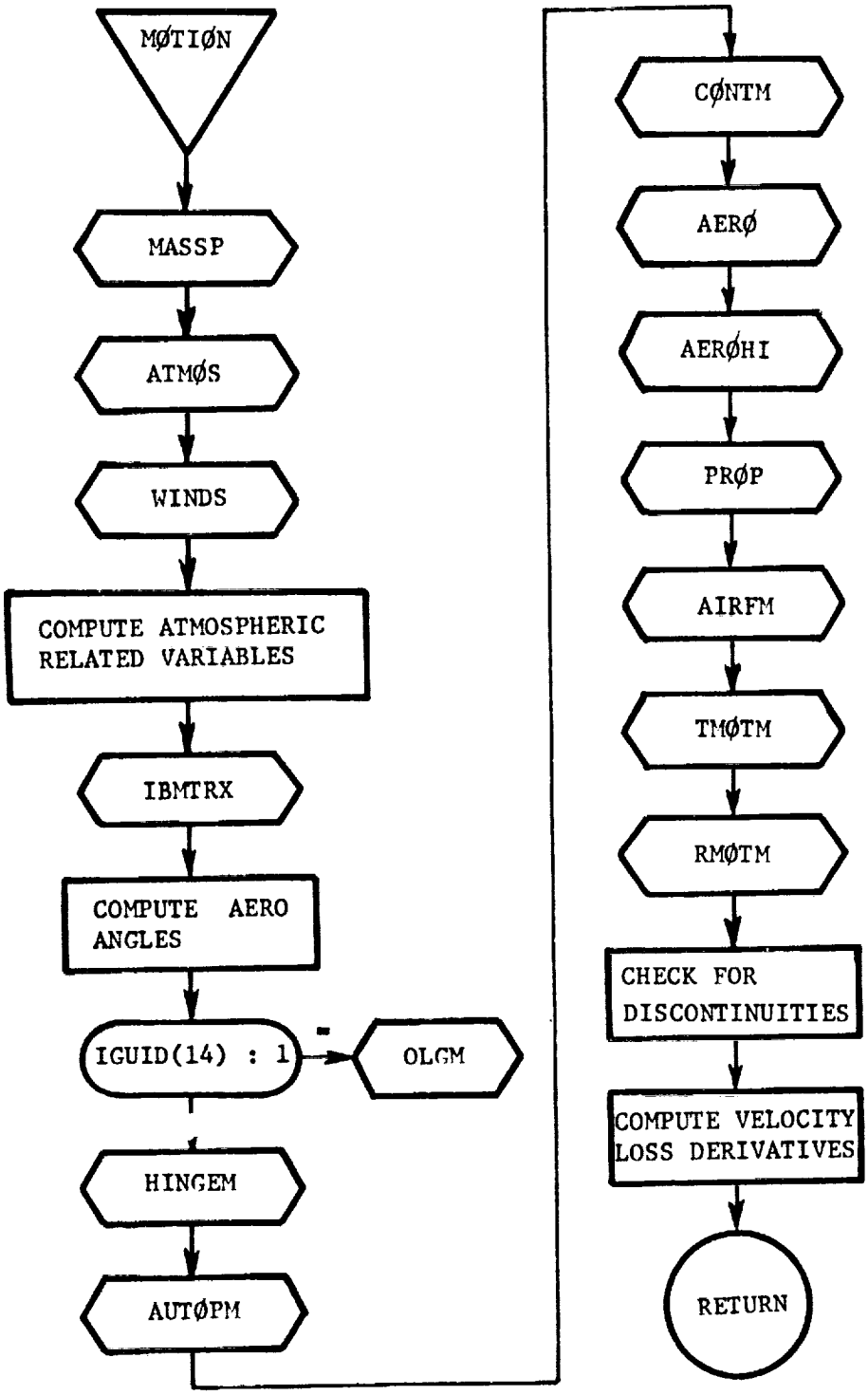
MØNITR: This routine determines the maximum and minimum values of the user-specified monitor variables.



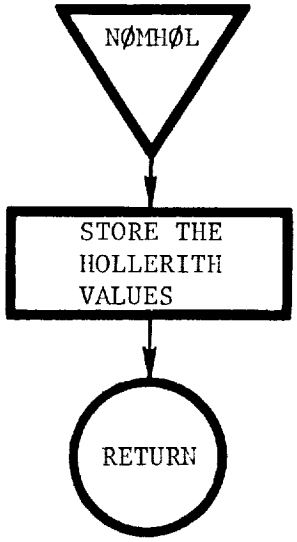
MØTIAL: This routine initializes the equations of motion.



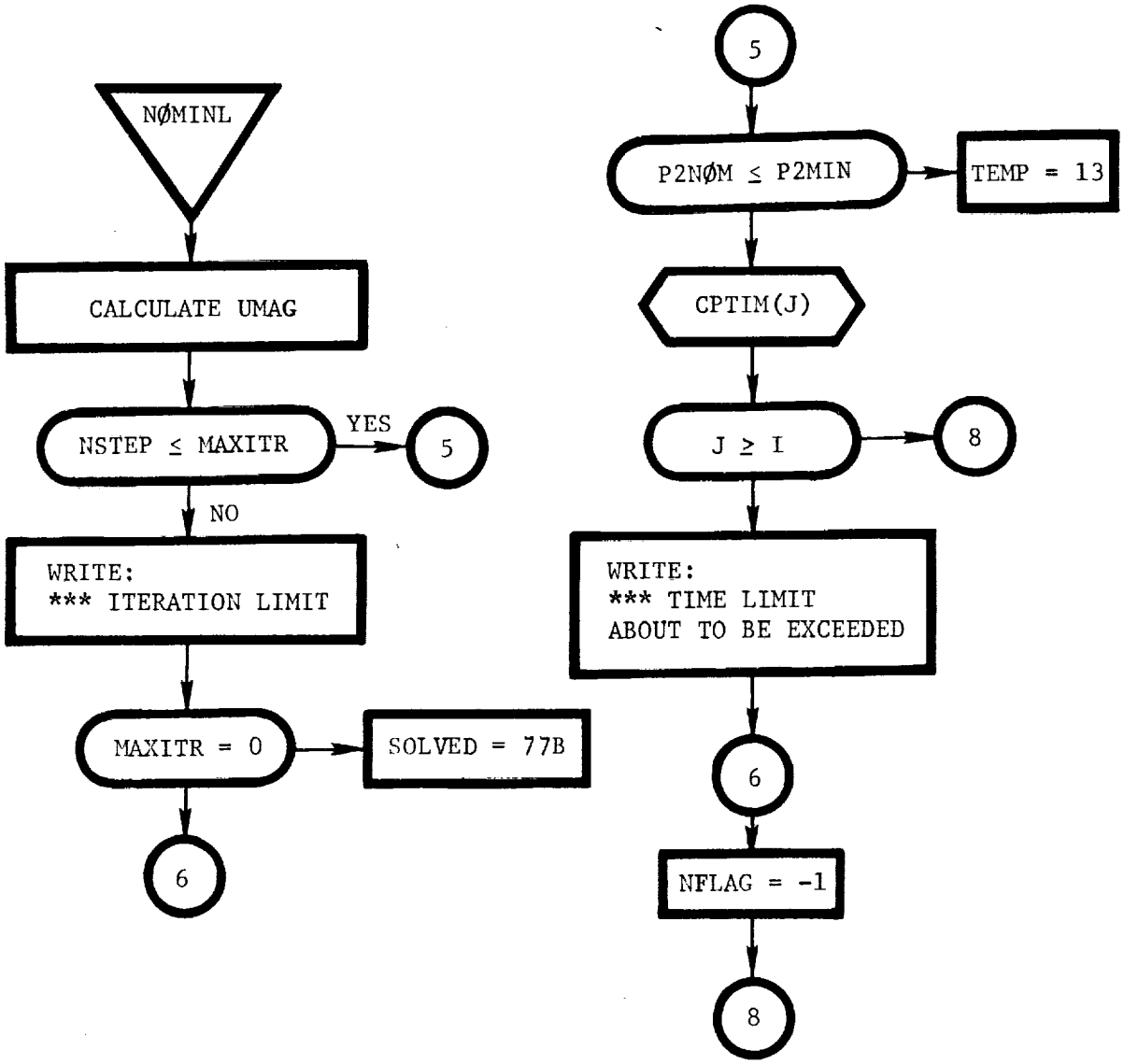
MØTIØN: This routine calculates the derivatives of the equations of motion.

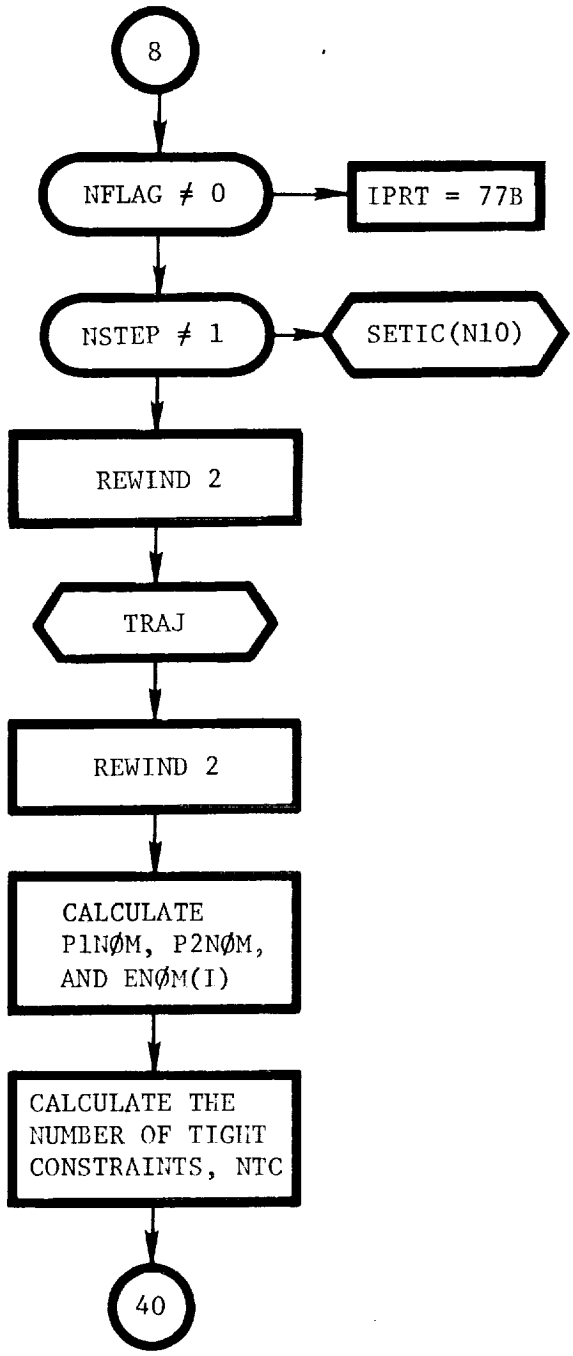


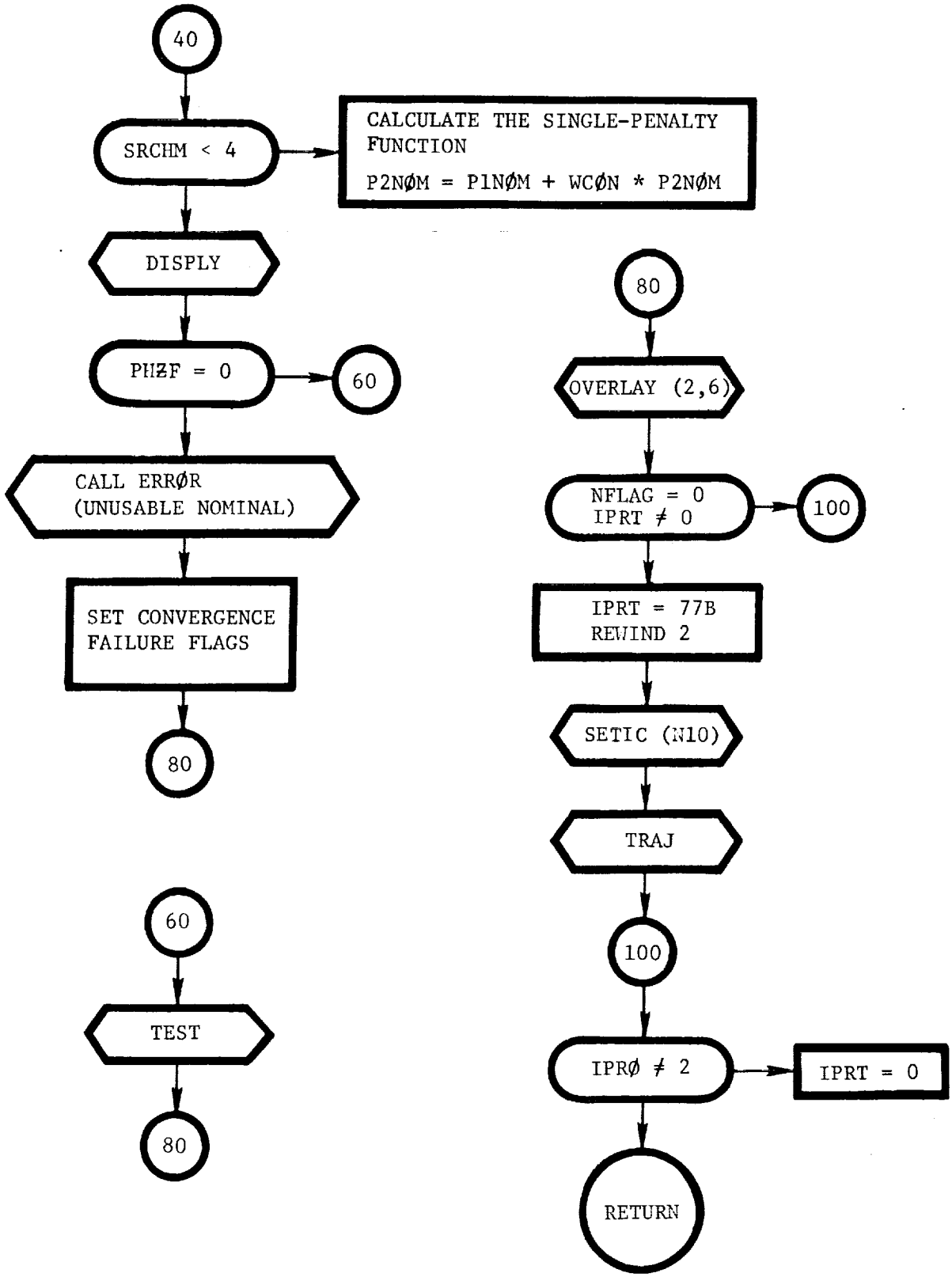
NØMHØL: This routine initializes the values of all HOLLERITH variables to the stored values.



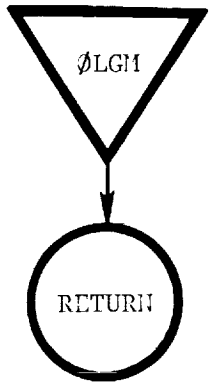
NØMINL: This subroutine runs the nominal trajectories (one per iteration), saving core at the beginning of each phase. The routine also calls TEST to determine if the iteration reference has converged or failed to converge.



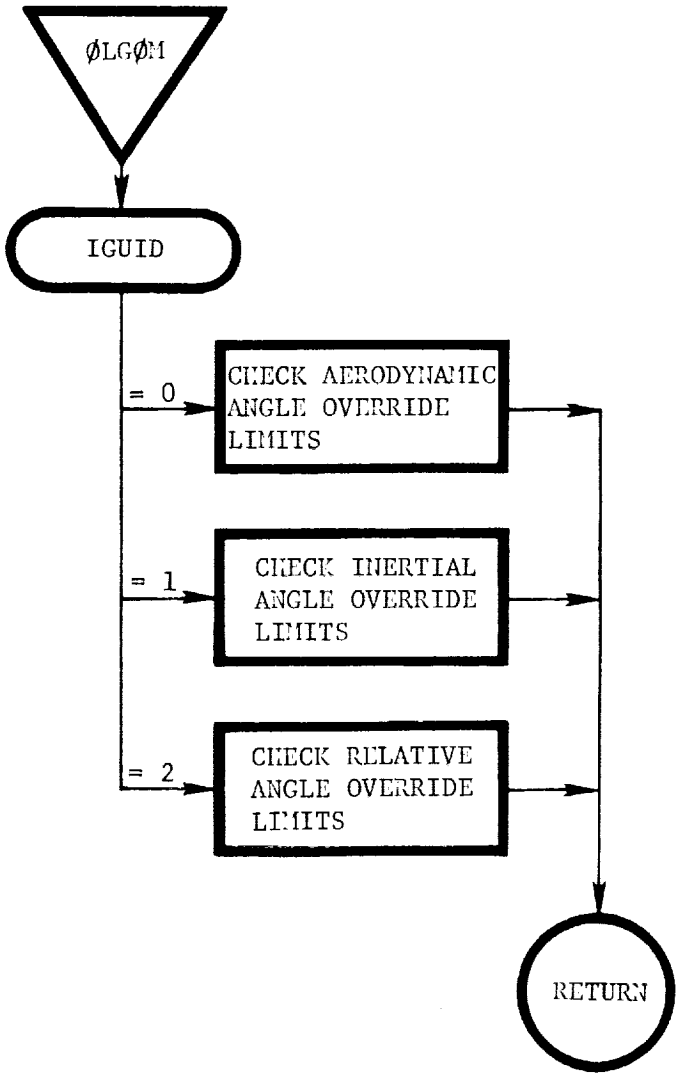

8
NFLAG ≠ 0
IPRT = 77B
NSTEP ≠ 1
SETIC(N10)
REWIND 2
TRAJ
REWIND 2
CALCULATE P1NØM, P2NØM, AND ENØM(I)
CALCULATE THE NUMBER OF TIGHT CONSTRAINTS, NTC
40


40
SRCHM < 4
CALCULATE THE SINGLE-PENALTY FUNCTION
P2NØM = P1NØM + WCØN * P2NØM
DISPLY
PHZF = 0
60
CALL ERRØR
(UNUSABLE NOMINAL)
SET CONVERGENCE FAILURE FLAGS
80
60
TEST
80
80
OVERLAY (2,6)
NFLAG = 0
IPRT ≠ 0
100
IPRT = 77B
REWIND 2
SETIC (N10)
TRAJ
100
IPRØ ≠ 2
IPRT = 0
RETURN

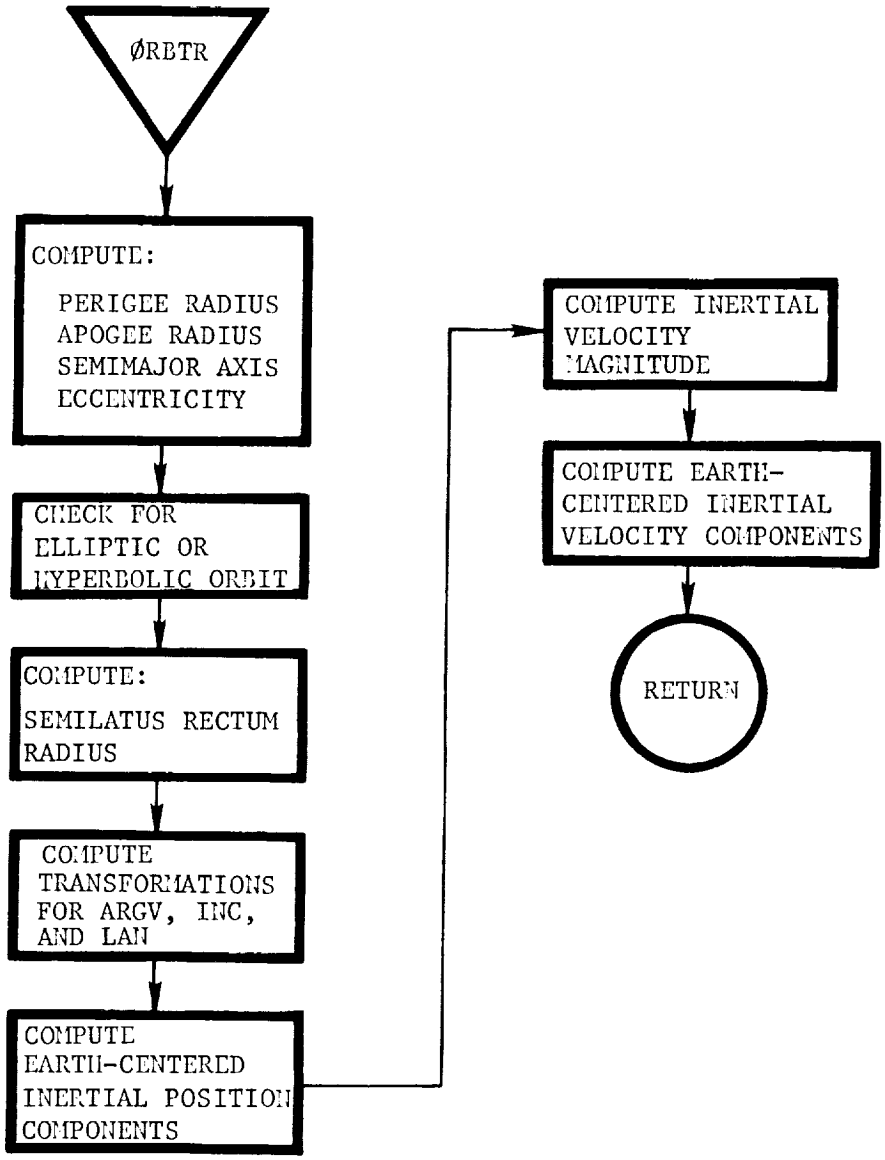
ØLGM: This is a blank routine that is to be used for special open-loop guidance (steering) models. The polynominal coefficients or angular values to be used by GUID1 can be calculated in this routine and then used by the user-selected option, based on the IGUID array.



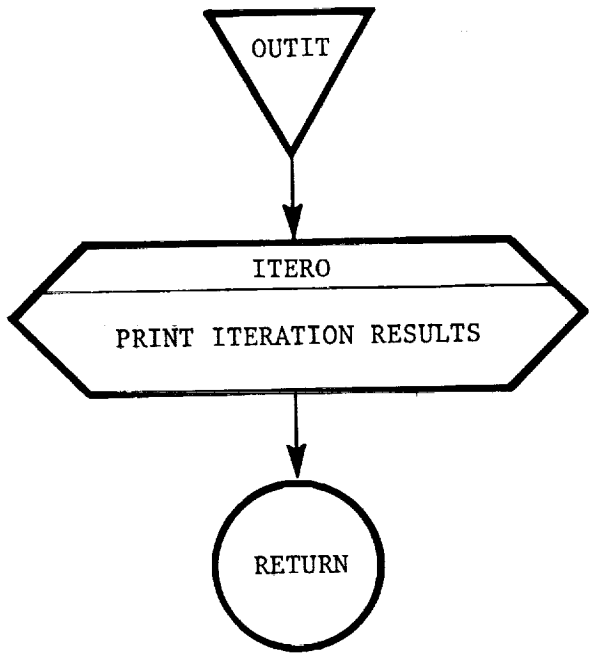
ØLGØM: This routine allows the guidance (steering) option values determined by the IGUID array to be overridden if a specified parameter test has been violated by the commanded angle. For example, the commanded angle of attack can be overridden if the value of QALPHA exceeds an input limit. This allows the program to follow the limit until it is no longer violated by the commanded angle.



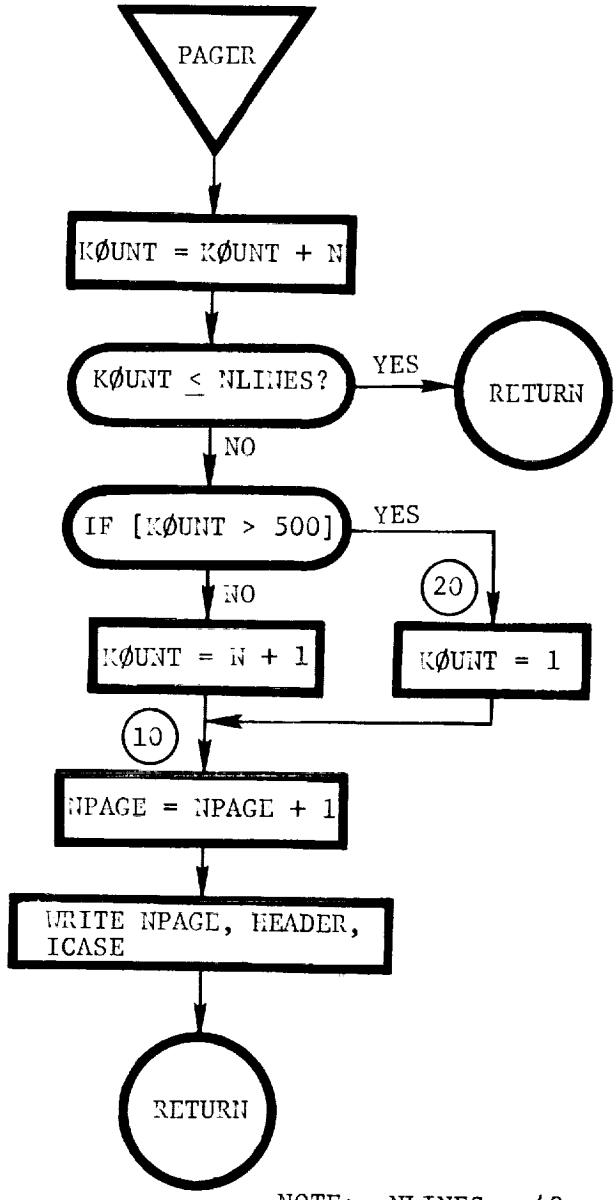
ØRBTR: This routine performs the transformation from orbital parameters to rectangular coordinates.



OUTIT: Program of overlay (2, 6). Its function is merely to call itero, which prints the iteration results.

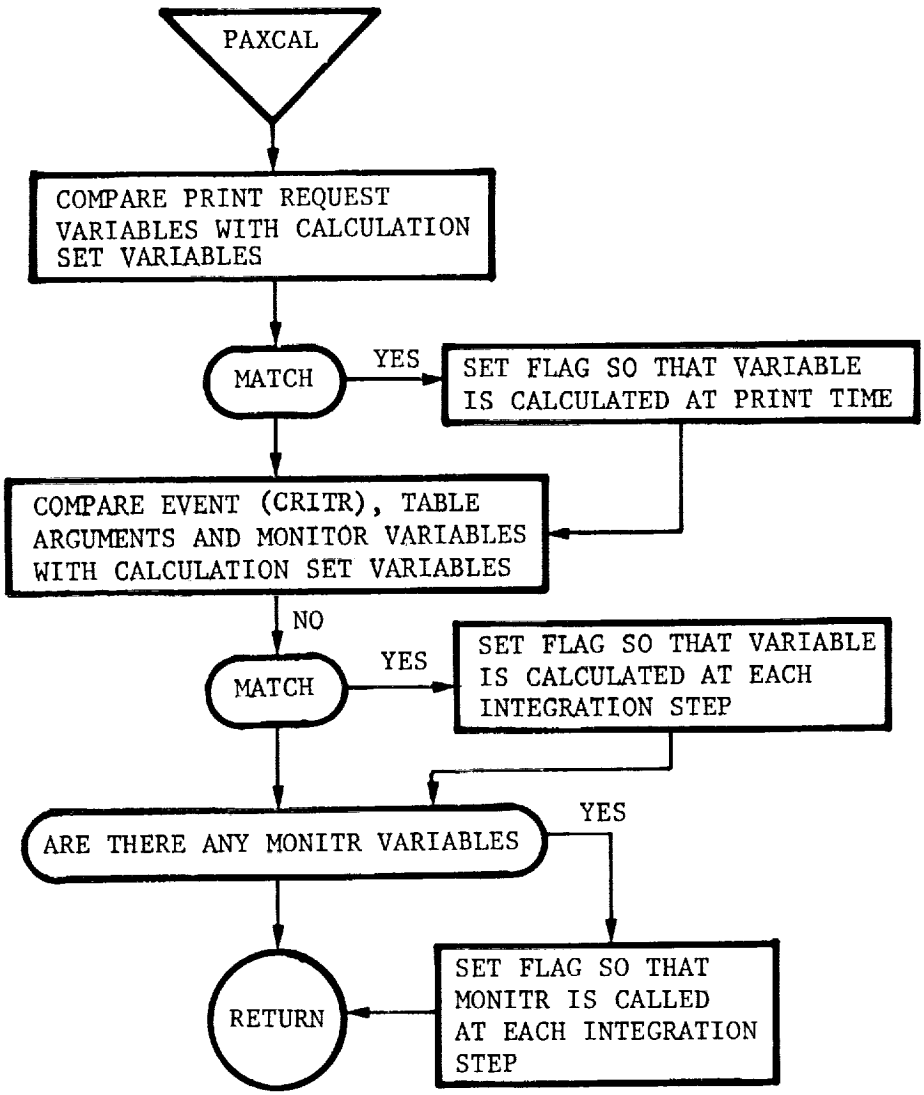


PAGER(N): This routine determines when a new page is required prior to printing. The argument is the number of lines to be printed.

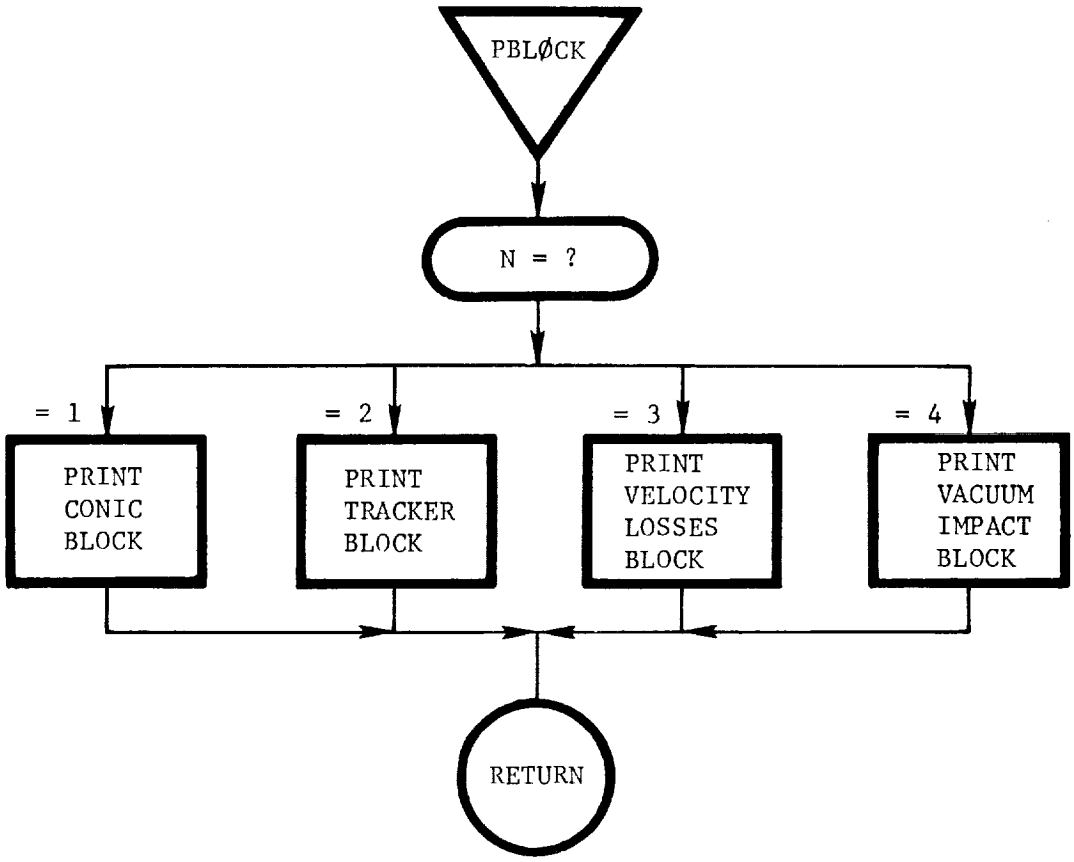


NOTE: NLINES = 48

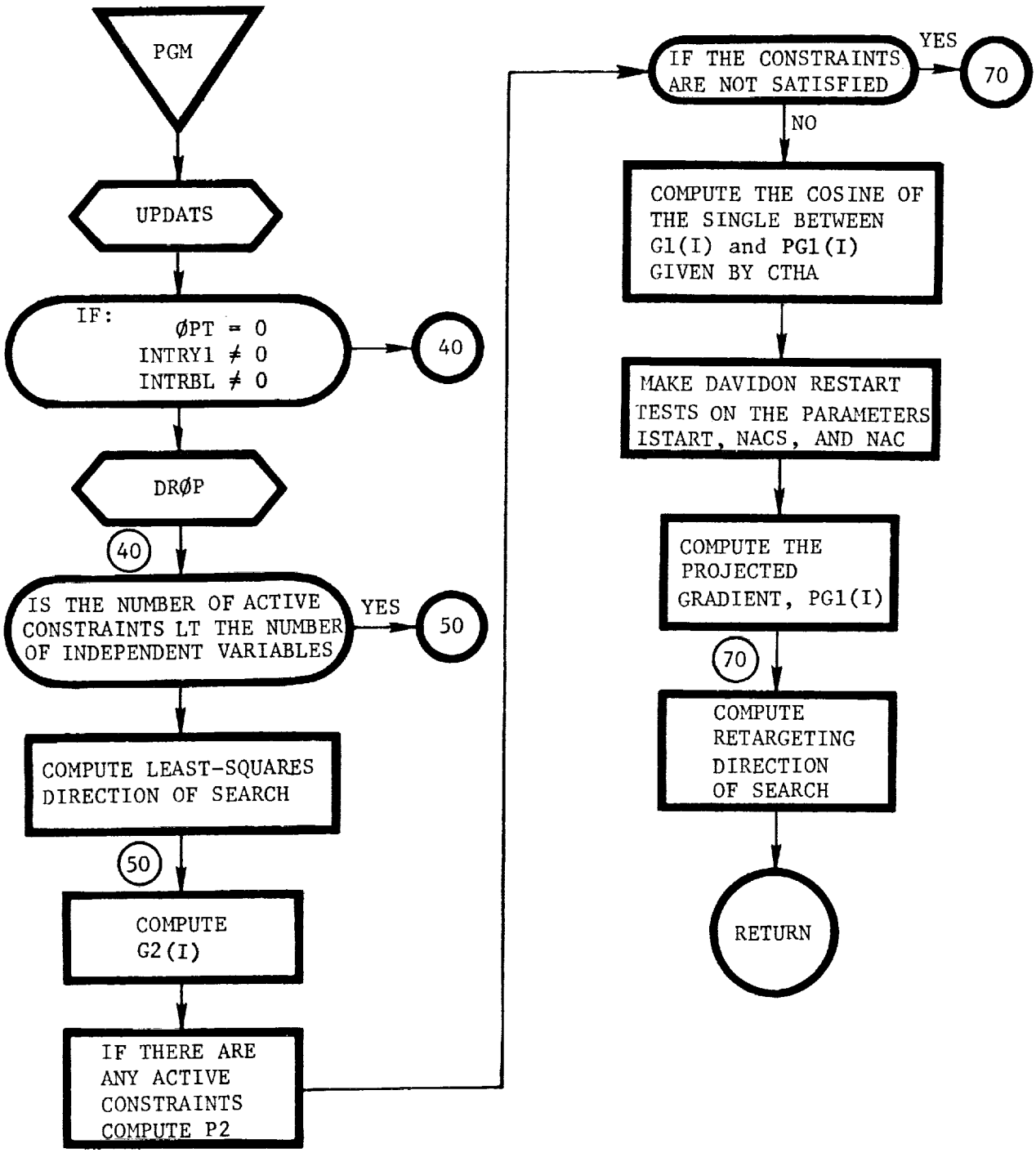
PAXCAL: This routine determines which variables need to be calculated and how often in the routines AUXFM, CONIC, MONITR and CALSPE.



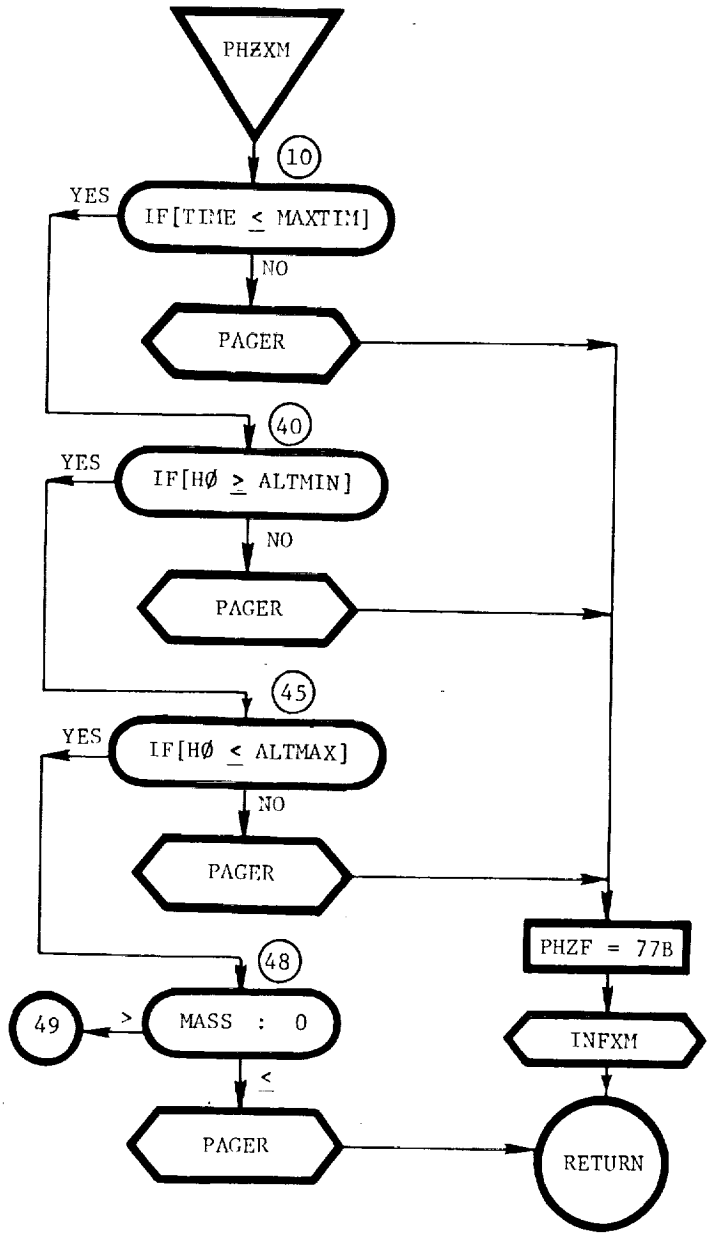
PBLØCK (N): This routine generates a summary print block for the option requested by the argument N.

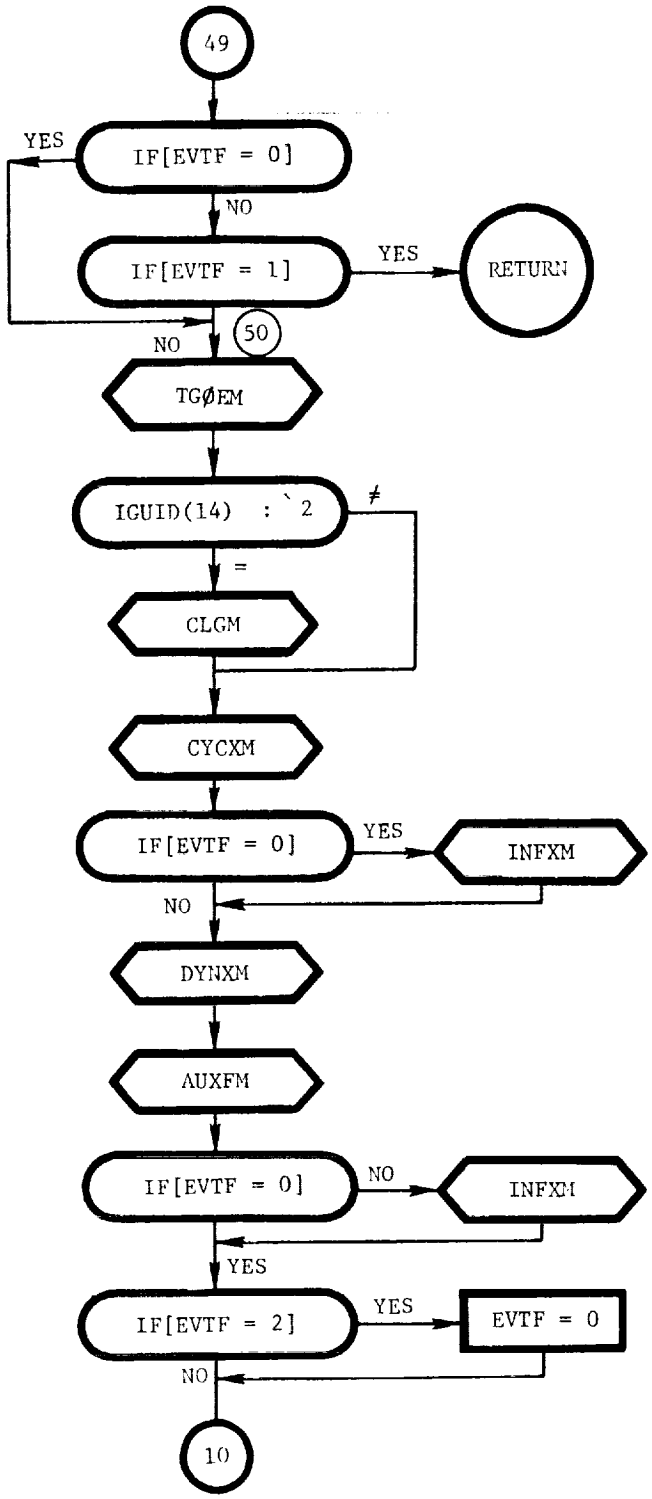


PGM: This routine determines the direction of search when the projected gradient method is used.

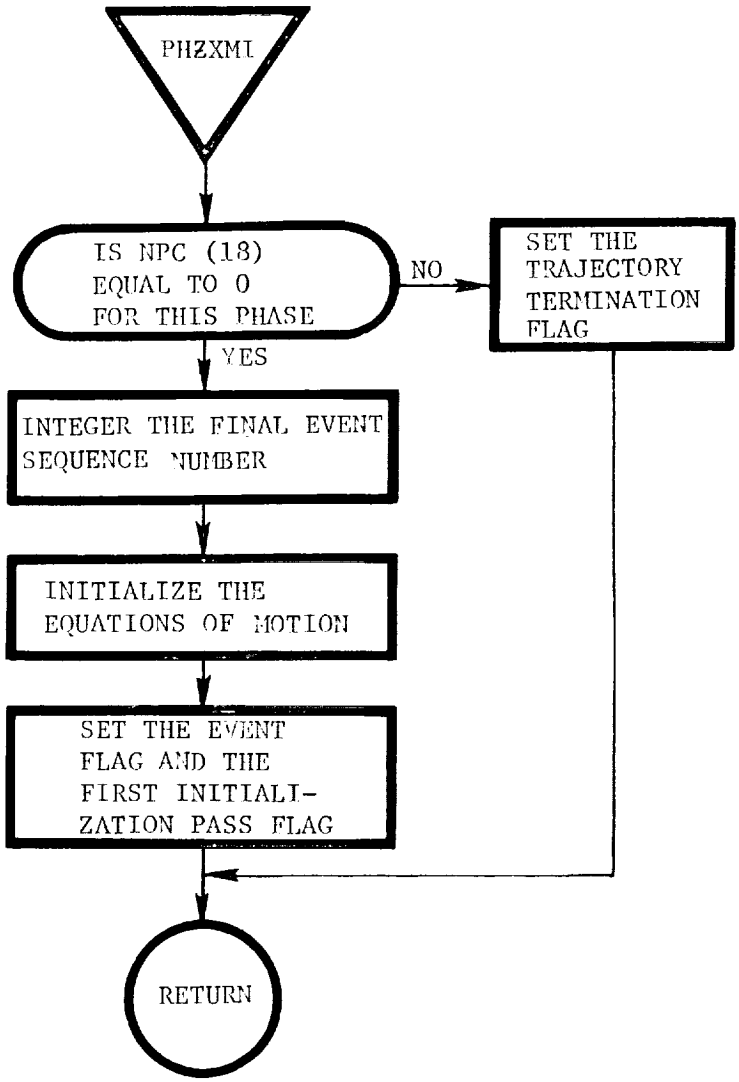


PHZXM: This routine is the executive routine for overlay (2,3). It controls the integration of the equations of motion and determines whether the parameters MAXTIM, ALTMIN, or ALTMAX have been exceeded; if they have, the trajectory is terminated.

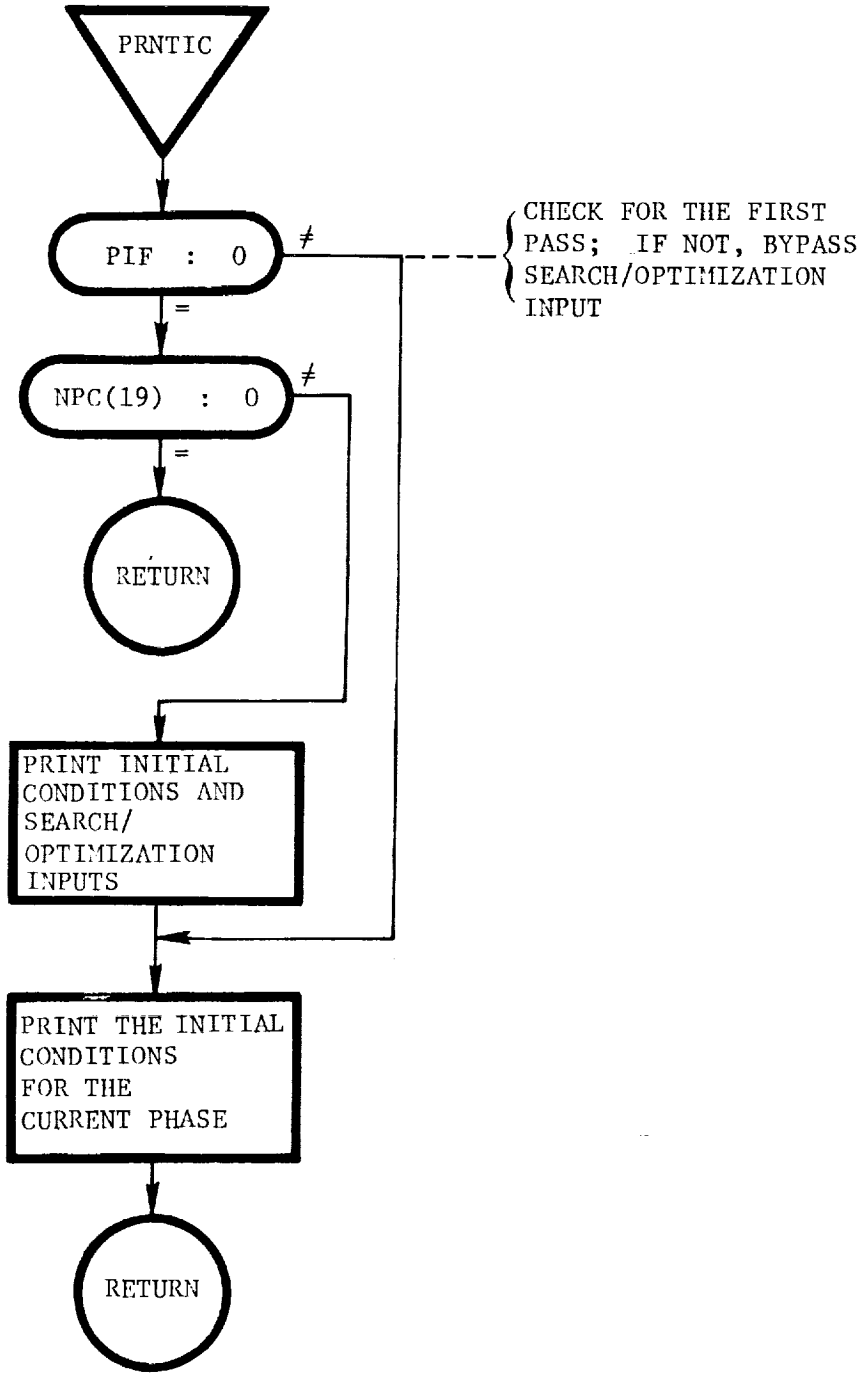



49
YES
IF[EVTF = 0]
NO
IF[EVTF = 1]
YES
RETURN
NO
50
TGØEM
IGUID(14) : `2
≠
=
CLGM
CYCXM
IF[EVTF = 0]
YES
INFXM
NO
DYNXM
AUXFM
IF[EVTF = 0]
NO
INFXM
YES
IF[EVTF = 2]
YES
EVTF = 0
NO
10

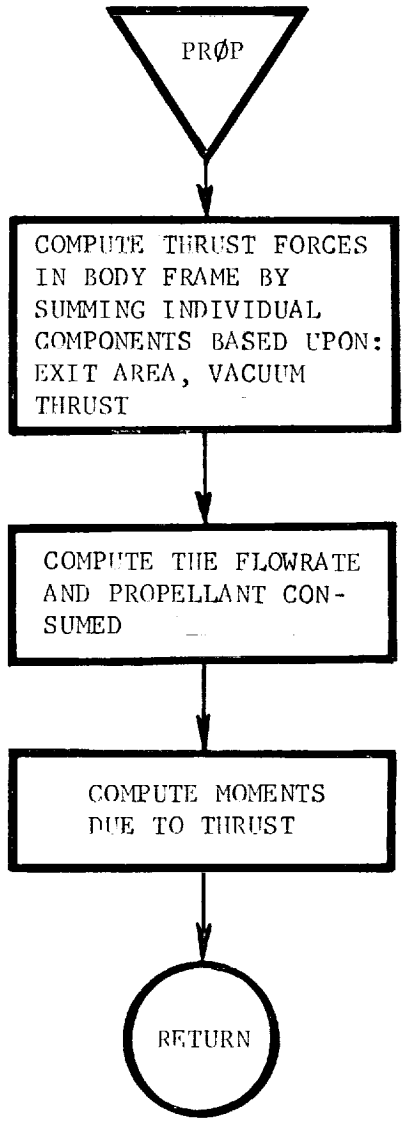
PHZXMI: This routine is the executive program of overlay (2,2). It performs the executive function for the phase initialization process.



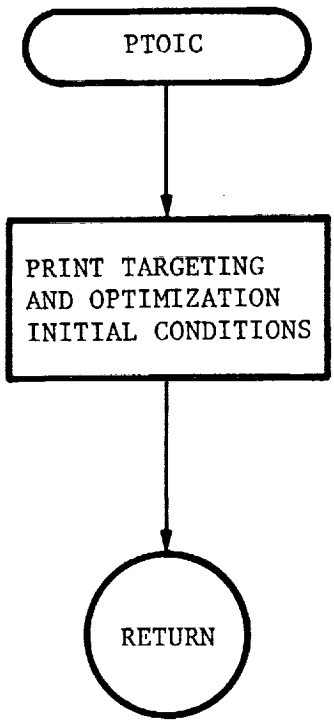
PRNTIC: This routine prints a summary of the initial conditions for the current phase.



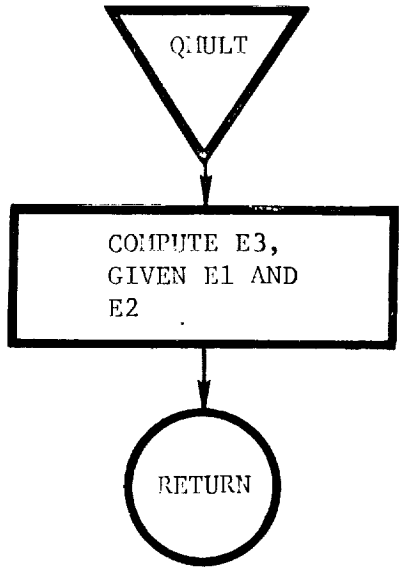
PRØP: This routine calculates the thrust forces and moments for rocket or jet engines. These forces and moments are computed in the body frame.



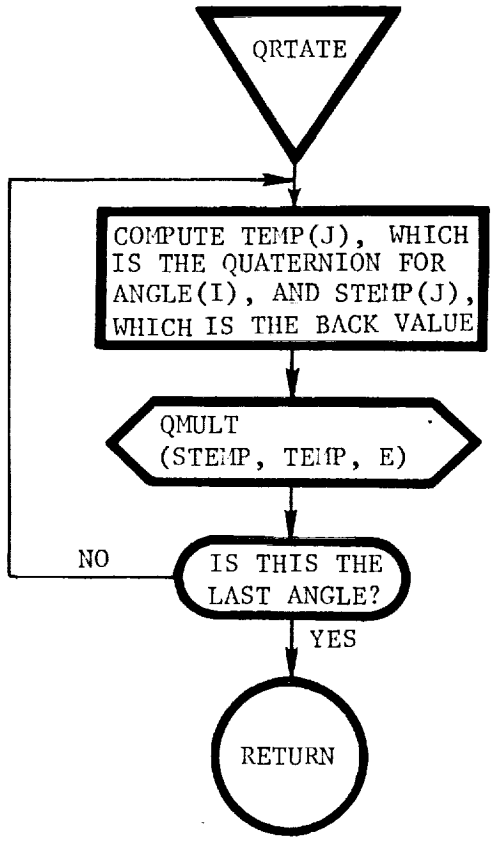
PTOIC: This routine prints the targeting and optimization initial conditions.



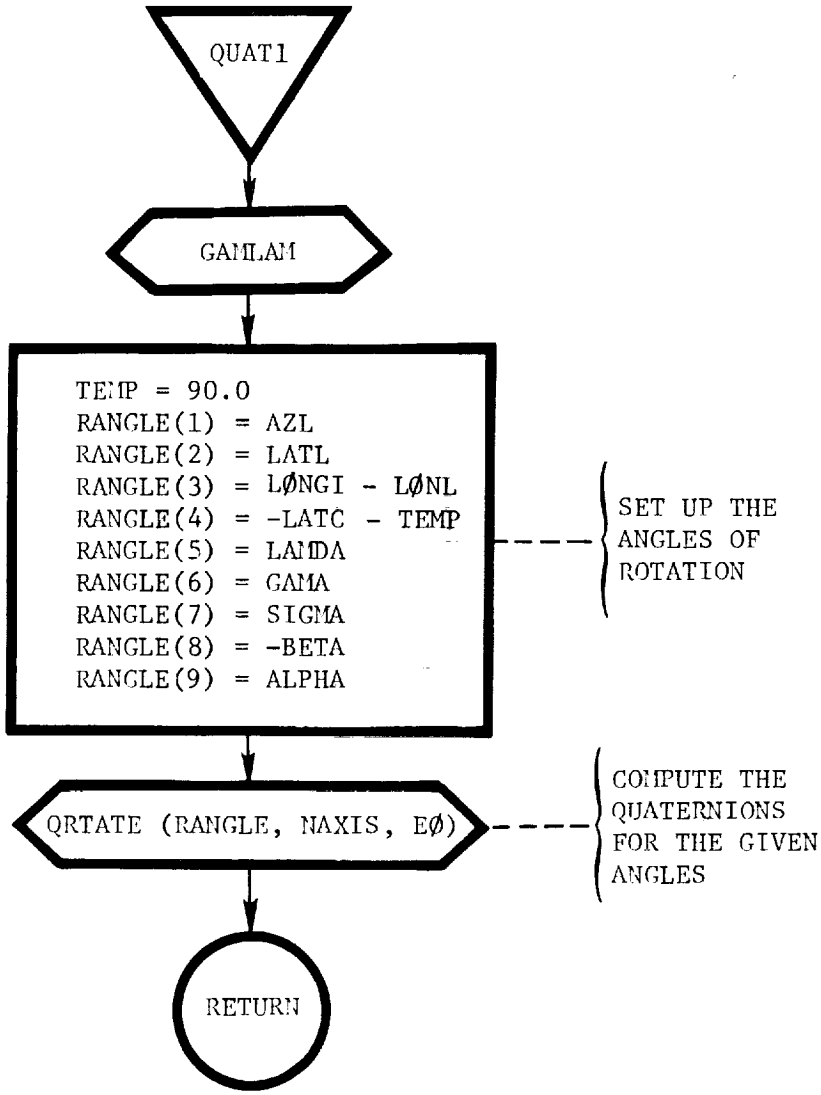
QMULT: This routine multiplies quaternion E1 and quaternion E2 to yield quaternion E3.



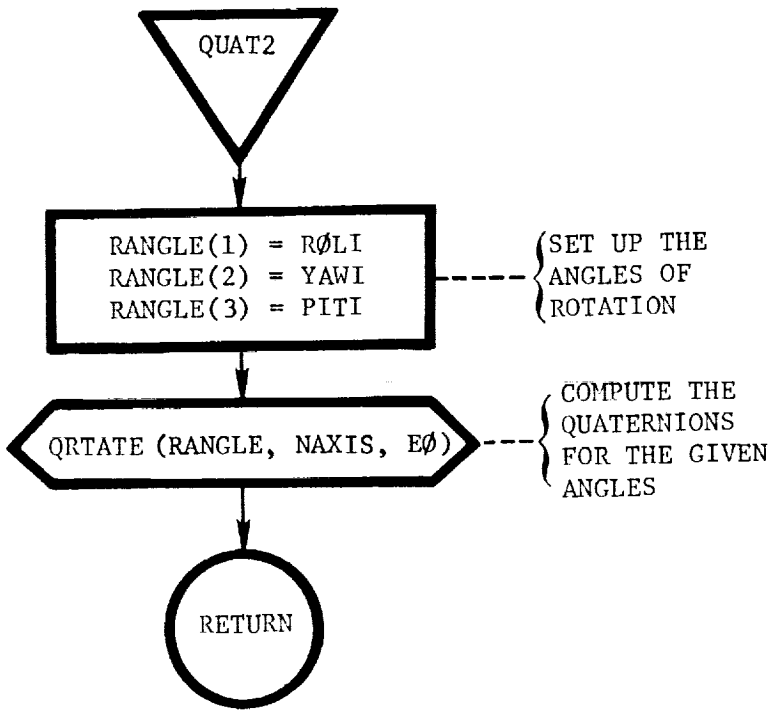
QRTATE: This routine computes the quaternion E that results from rotating through the specified angles, ANGLE, about the specified axes, NAXIS.



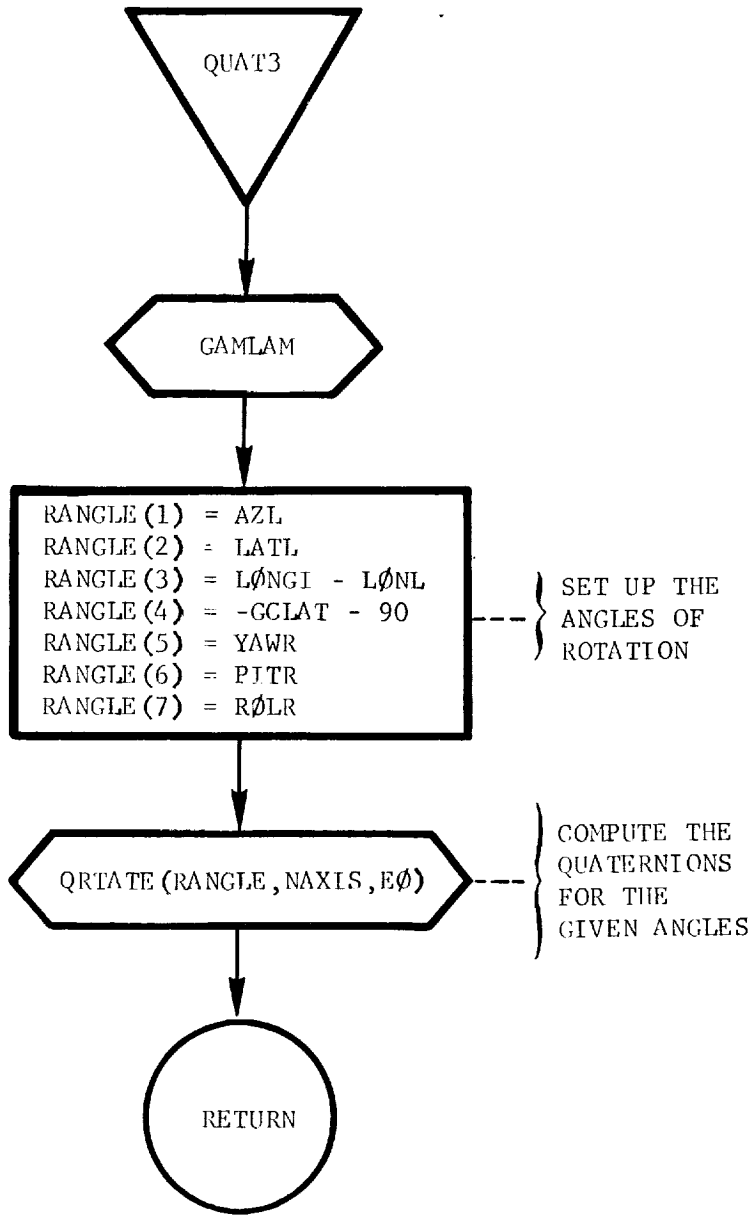
QUAT1: This routine computes the quaternions, given the angle of attack, sideslip, and bank angle.



QUAT2: This routine computes the quaternions, given the local attitude angles in yaw, pitch, and roll.



QUAT3: This routine computes the quaternions given the relative euler angles.



READAT: This routine performs the actual processing of the input data. Subroutines RSEARCH, RGENDAT, RTBLMLT, and RTAB are called as required to perform the actual reading of namelists "SEARCH", "GENDAT", "TBLMLT", and "TAB", respectively. The namelists are always read in a given sequence that can be terminated at any point by setting ENDPHS = 1. The calling sequence for reading the namelists is: RSEARCH, RGENDAT, RTBLMLT, and RTAB.

After reading each namelist, the data for that phase are packed into one of two data buffers, depending on the type of data being processed. The two data buffers are broken down as follows:

1) IGEN: 1500 decimal cells of storage for all input variables except for event criteria and input tables for all phases. The table multipliers are also stored in this array.

2) IBKT: 1500 decimal cells of storage for the event criteria and tables for all phases.

The actual storage detail is shown in Table 2. The input variables are stored in sequence as they are read and the data are grouped according to phase number.

TABLE III-1

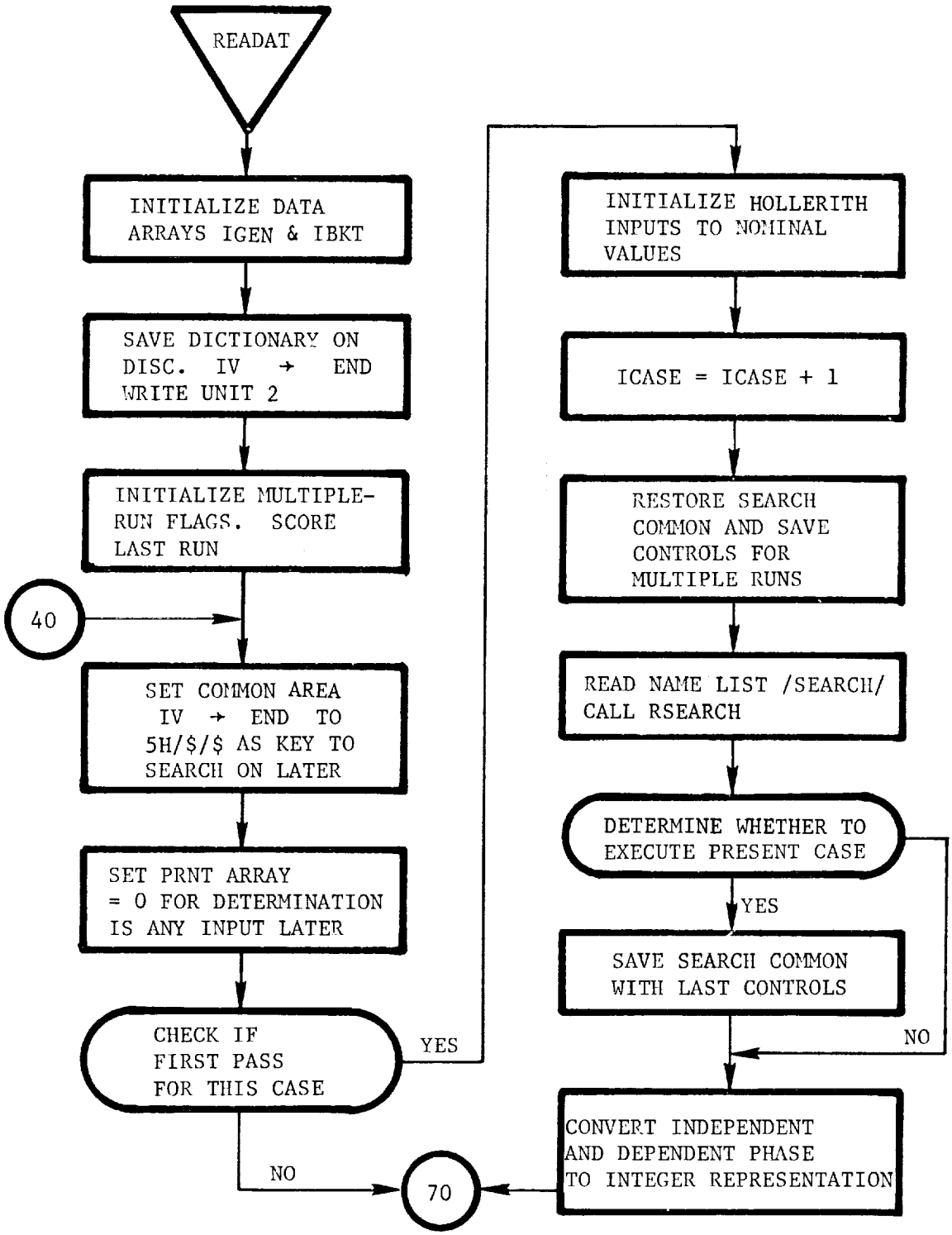
CONTENTS OF DATA BUFFERS

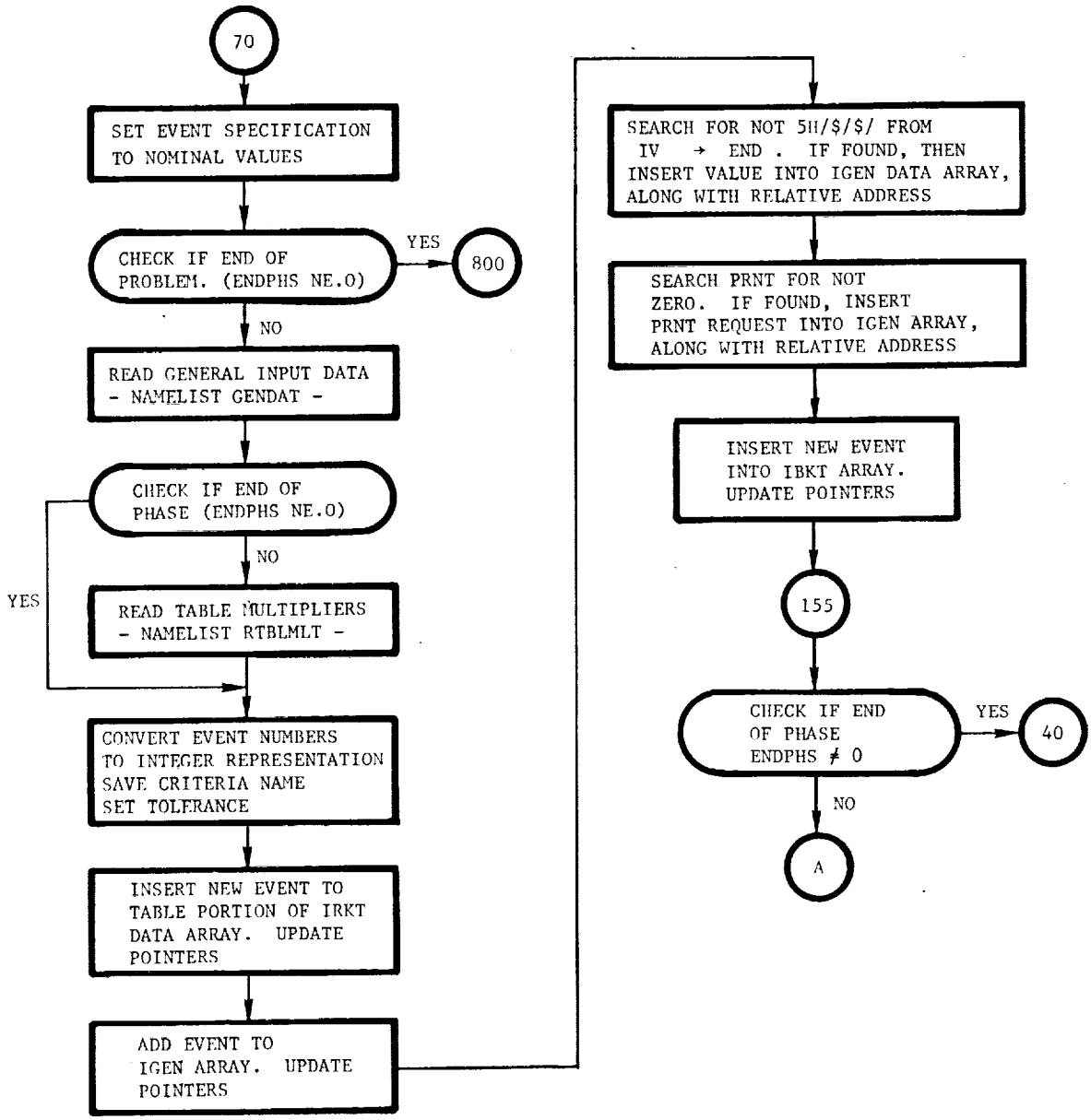
Location	BUFFER "IGEN" (1500 decimal cells)	Location	BUFFER "IBKT" (1500 decimal cells)
1	number of cells occupied in IGEN	1	number of cells occupied in IBKT
2	number of cells occupied by data for the first phase (NG1)	2	number of cells occupied by event sequence data (NB1)
3	event sequence number for the first phase	3	event sequence number for the first phase
4	address of the first variable following the event sequence number in IGEN	4	event type for the first phase
5	value of the first variable in IGEN	5	event criteria for the first phase
6	unused cell associated with the first variable stored in IGEN	6	criteria value for the first phase
7 thru NG1	variables stored by repeating the sequence shown in 4, 5, 6 for each variable	7	derivative name of the event criteria variable for the first phase
		8	tolerance
NG1+1	number of cells occupied by data for the second phase (NG2)	9	model number
NG1+2	repeat sequence 3 through 7-NG2 as before for the second event	10	unused cell associated with the event criteria for the first event
		11 thru NB1	repeat sequence 3-10 for remaining events

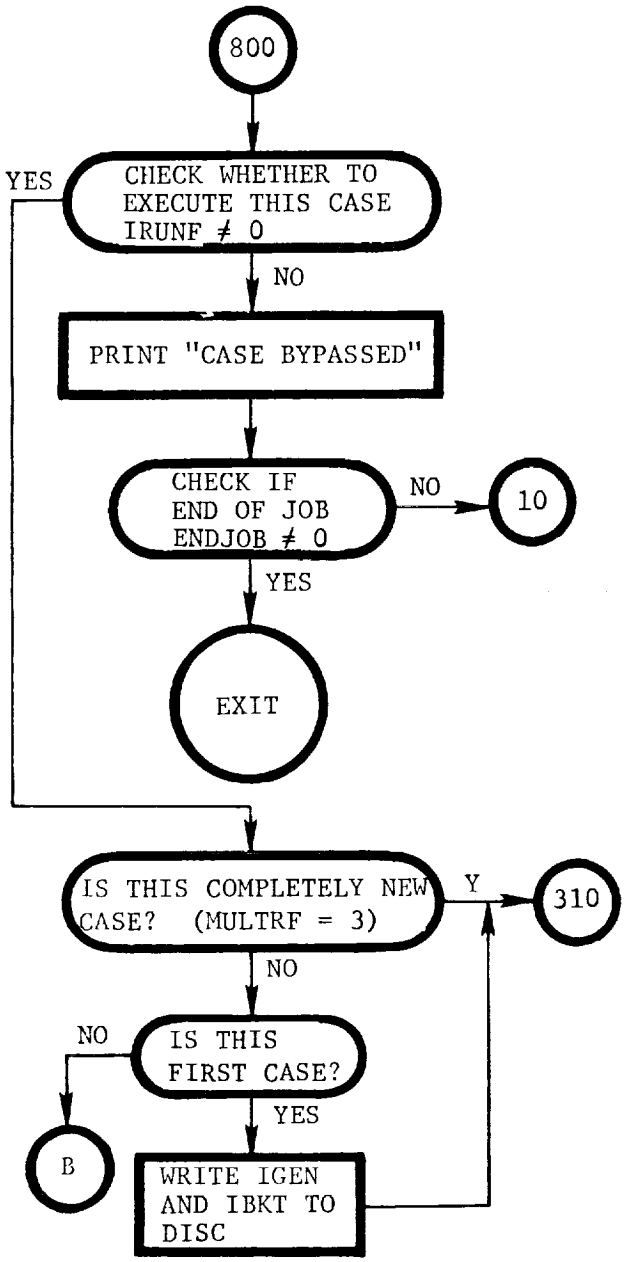
TABLE III-1

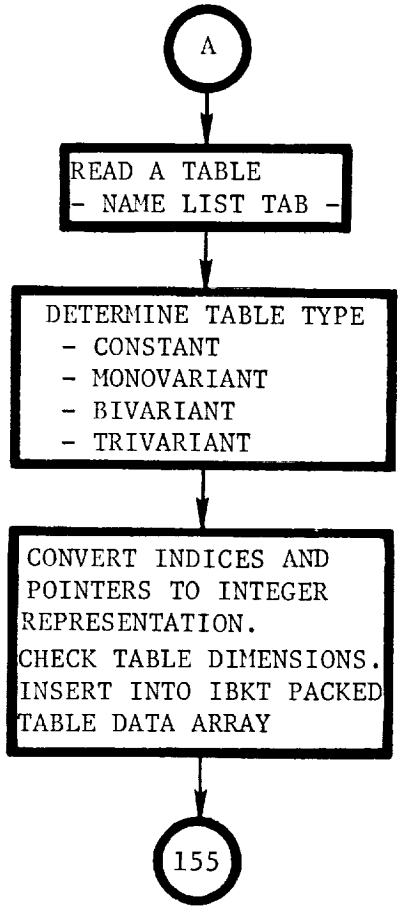
CONTENTS OF DATA BUFFERS - CONCLUDED

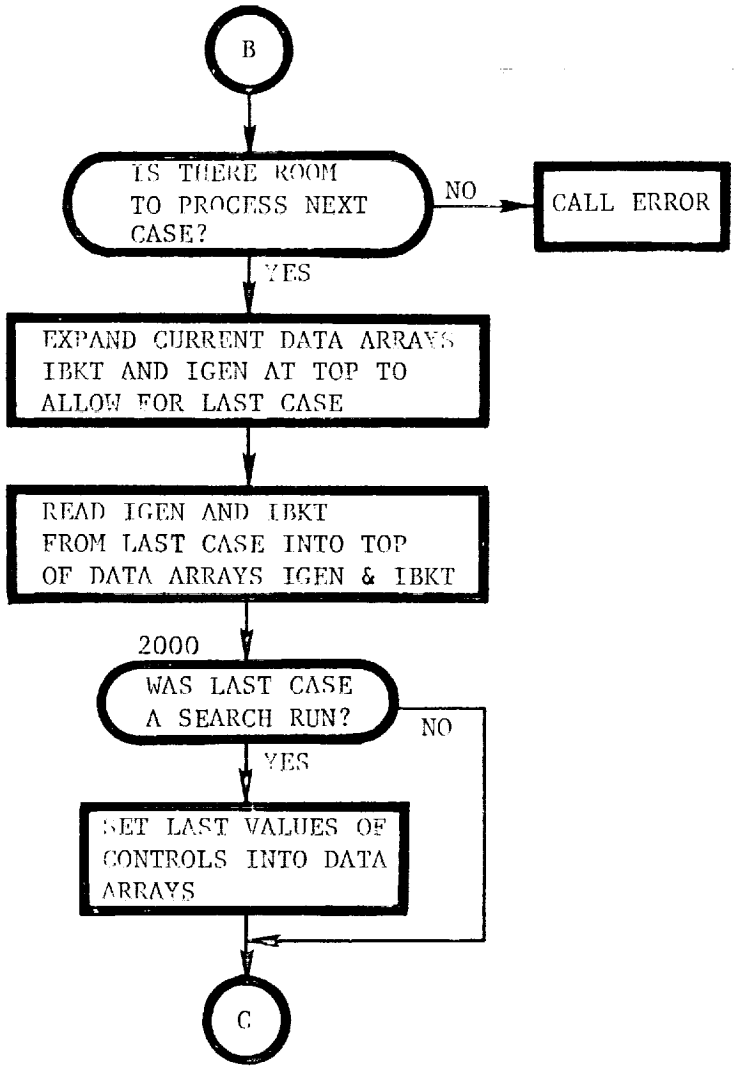
Location	BUFFER "IGEN"	Location	BUFFER "IBKT"
		NB1 + 1	number of cells occupied by all tables for the first phase (NB2)
		NB1 + 2	phase number associated with the first set of tables
		NB1 + 3	size of the first table to be input
		NB1 + 4	name of the first table (HOLLERITH)
		NB1 + 5	table pointers and values
			repeat above sequence for each table in the first phase (NB1 + 3 thru NB1 + 5)
			repeat above sequence for all phases (NB1 + 1 thru NB1 + 5)

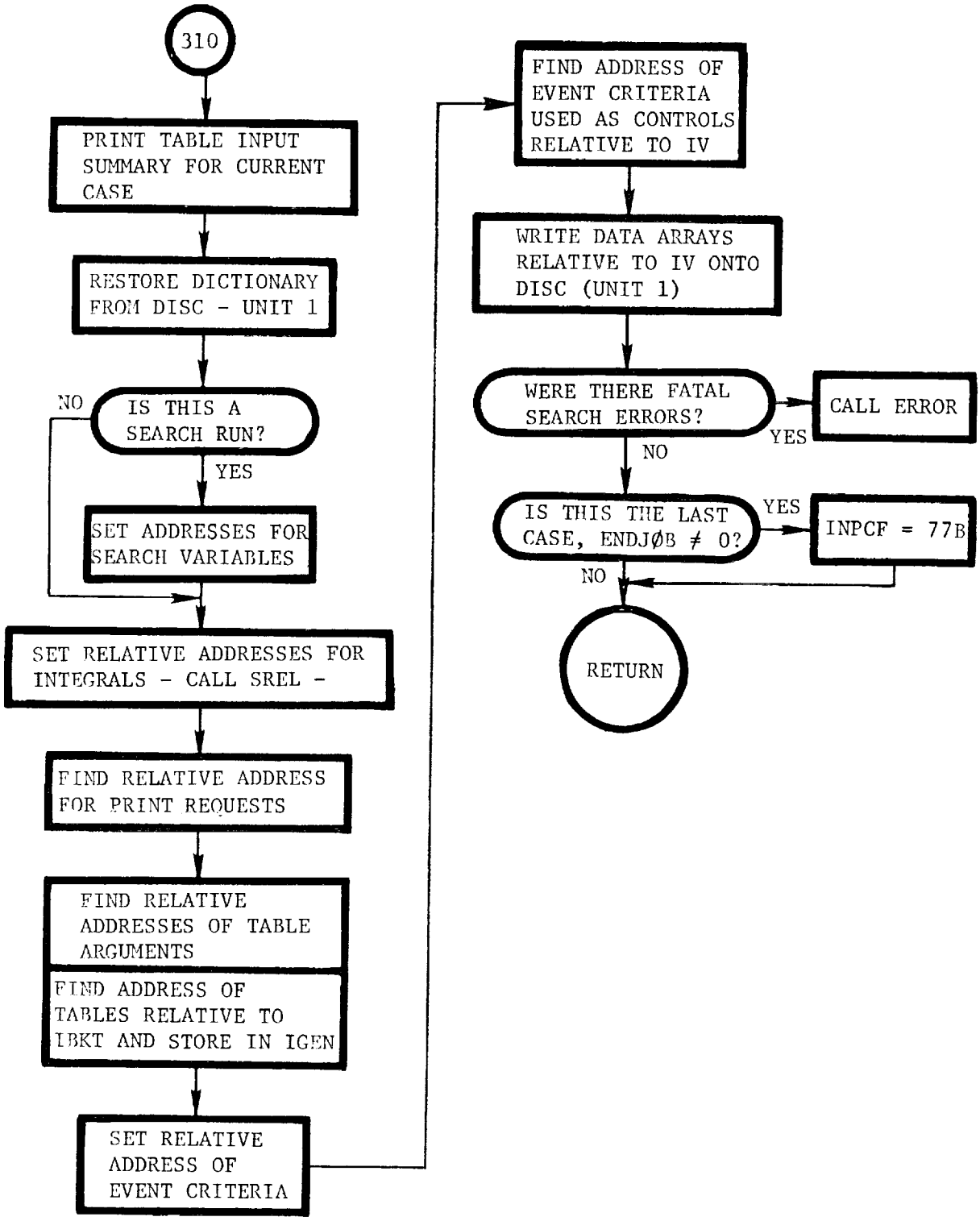

READAT
INITIALIZE DATA ARRAYS IGEN & IBKT
SAVE DICTIONARY ON DISC. IV → END WRITE UNIT 2
INITIALIZE MULTIPLE-RUN FLAGS. SCORE LAST RUN
40
SET COMMON AREA IV → END TO 5H/$/$ AS KEY TO SEARCH ON LATER
SET PRNT ARRAY = 0 FOR DETERMINATION IS ANY INPUT LATER
CHECK IF FIRST PASS FOR THIS CASE
YES
NO
70
INITIALIZE HOLLERITH INPUTS TO NOMINAL VALUES
ICASE = ICASE + 1
RESTORE SEARCH COMMON AND SAVE CONTROLS FOR MULTIPLE RUNS
READ NAME LIST /SEARCH/ CALL RSEARCH
DETERMINE WHETHER TO EXECUTE PRESENT CASE
YES
NO
SAVE SEARCH COMMON WITH LAST CONTROLS
CONVERT INDEPENDENT AND DEPENDENT PHASE TO INTEGER REPRESENTATION

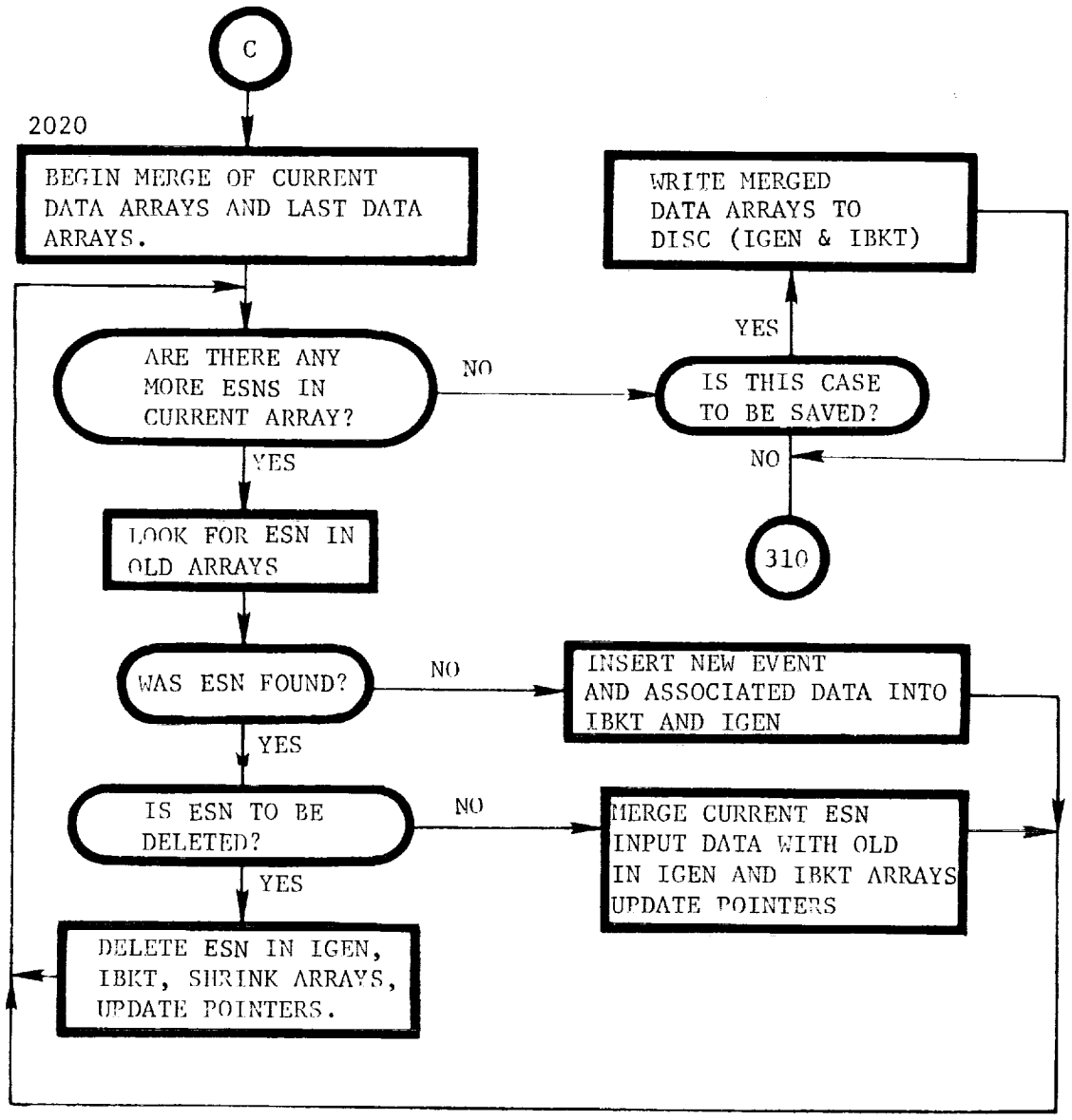

70
SET EVENT SPECIFICATION TO NOMINAL VALUES
CHECK IF END OF PROBLEM. (ENDPHS NE.0)
YES
800
NO
READ GENERAL INPUT DATA - NAMELIST GENDAT -
CHECK IF END OF PHASE (ENDPHS NE.0)
NO
YES
READ TABLE MULTIPLIERS - NAMELIST RTBLMLT -
CONVERT EVENT NUMBERS TO INTEGER REPRESENTATION SAVE CRITERIA NAME SET TOLERANCE
INSERT NEW EVENT TO TABLE PORTION OF IRKT DATA ARRAY. UPDATE POINTERS
ADD EVENT TO IGEN ARRAY. UPDATE POINTERS
SEARCH FOR NOT 5H/$/$/ FROM IV → END . IF FOUND, THEN INSERT VALUE INTO IGEN DATA ARRAY, ALONG WITH RELATIVE ADDRESS
SEARCH PRNT FOR NOT ZERO. IF FOUND, INSERT PRNT REQUEST INTO IGEN ARRAY, ALONG WITH RELATIVE ADDRESS
INSERT NEW EVENT INTO IBKT ARRAY. UPDATE POINTERS
155
CHECK IF END OF PHASE ENDPHS ≠ 0
YES
40
NO
A


800
YES
CHECK WHETHER TO EXECUTE THIS CASE IRUNF ≠ 0
NO
PRINT "CASE BYPASSED"
CHECK IF END OF JOB ENDJOB ≠ 0
NO
10
YES
EXIT
IS THIS COMPLETELY NEW CASE? (MULTRF = 3)
Y
310
NO
NO
IS THIS FIRST CASE?
YES
B
WRITE IGEN AND IBKT TO DISC

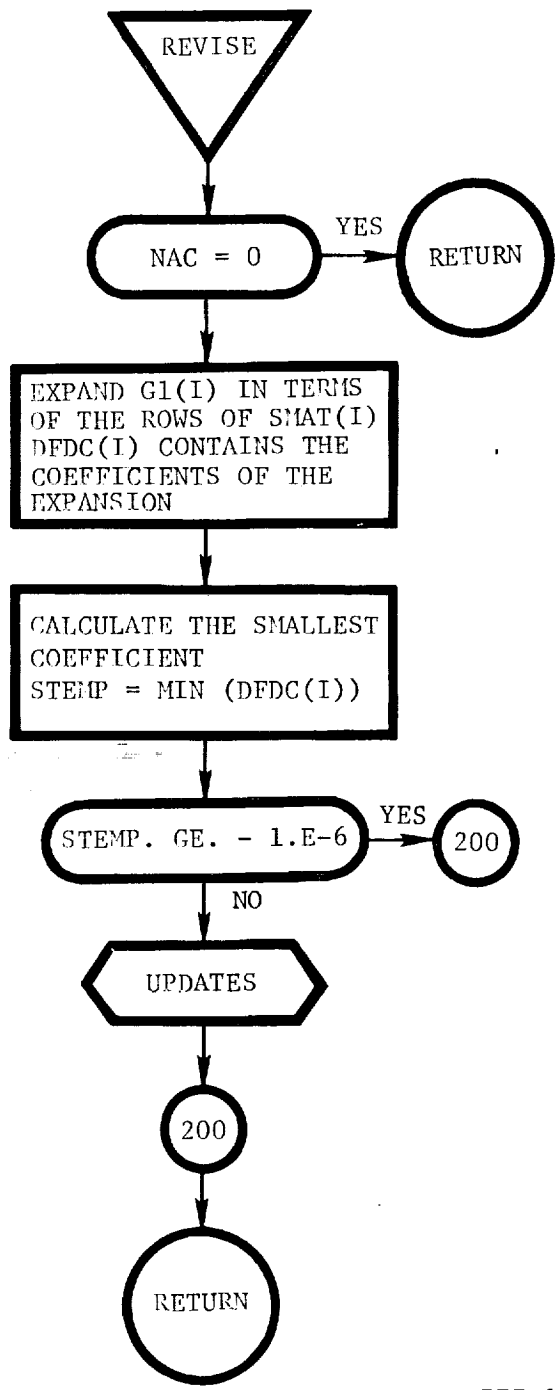

A
READ A TABLE
- NAME LIST TAB -
DETERMINE TABLE TYPE
- CONSTANT
- MONOVARIANT
- BIVARIANT
- TRIVARIANT
CONVERT INDICES AND POINTERS TO INTEGER REPRESENTATION.
CHECK TABLE DIMENSIONS.
INSERT INTO IBKT PACKED TABLE DATA ARRAY
155


B
IS THERE ROOM TO PROCESS NEXT CASE?
NO
CALL ERROR
YES
EXPAND CURRENT DATA ARRAYS IBKT AND IGEN AT TOP TO ALLOW FOR LAST CASE
READ IGEN AND IBKT FROM LAST CASE INTO TOP OF DATA ARRAYS IGEN & IBKT
2000
WAS LAST CASE A SEARCH RUN?
NO
YES
SET LAST VALUES OF CONTROLS INTO DATA ARRAYS
C

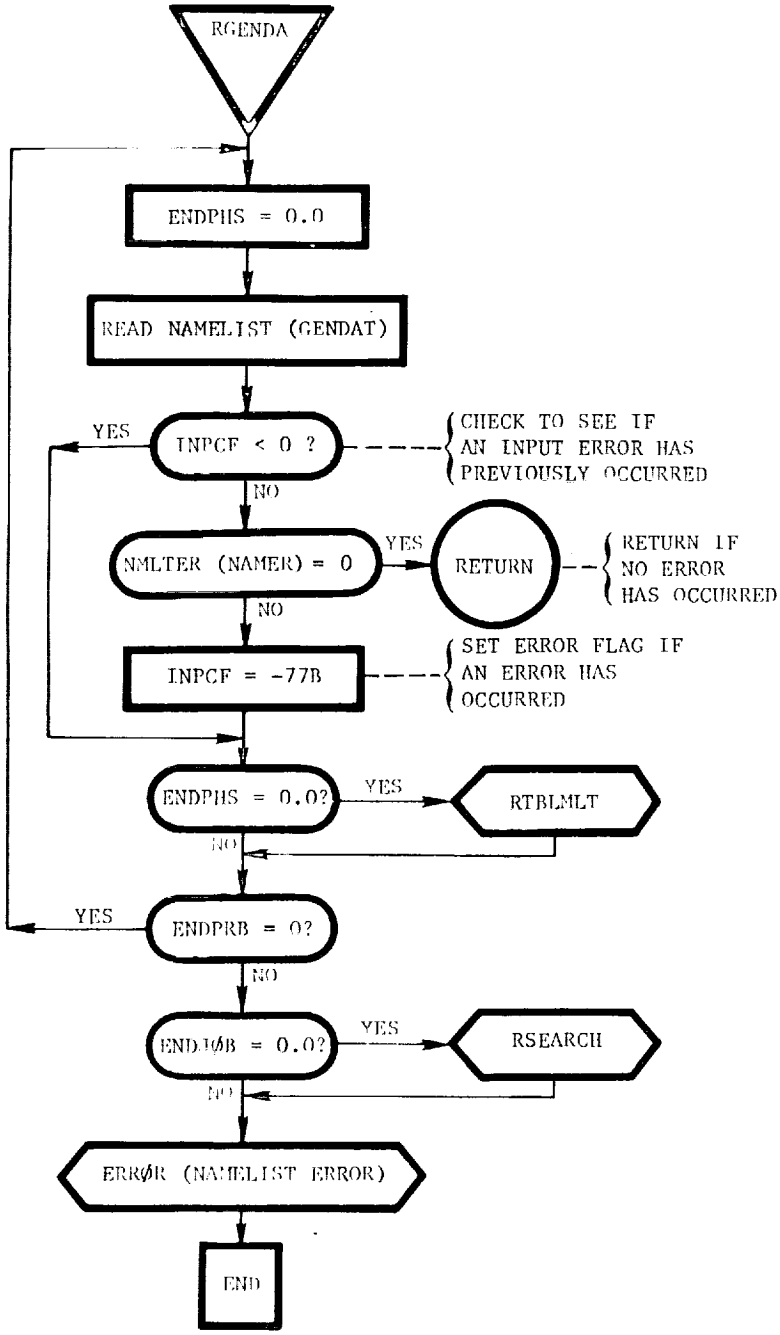

310
PRINT TABLE INPUT SUMMARY FOR CURRENT CASE
RESTORE DICTIONARY FROM DISC - UNIT 1
IS THIS A SEARCH RUN?
NO
YES
SET ADDRESSES FOR SEARCH VARIABLES
SET RELATIVE ADDRESSES FOR INTEGRALS - CALL SREL -
FIND RELATIVE ADDRESS FOR PRINT REQUESTS
FIND RELATIVE ADDRESSES OF TABLE ARGUMENTS
FIND ADDRESS OF TABLES RELATIVE TO IBKT AND STORE IN IGEN
SET RELATIVE ADDRESS OF EVENT CRITERIA
FIND ADDRESS OF EVENT CRITERIA USED AS CONTROLS RELATIVE TO IV
WRITE DATA ARRAYS RELATIVE TO IV ONTO DISC (UNIT 1)
WERE THERE FATAL SEARCH ERRORS?
YES
CALL ERROR
NO
IS THIS THE LAST CASE, ENDJØB ≠ 0?
YES
INPCF = 77B
NO
RETURN


C
2020
BEGIN MERGE OF CURRENT DATA ARRAYS AND LAST DATA ARRAYS.
ARE THERE ANY MORE ESNS IN CURRENT ARRAY?
NO
YES
LOOK FOR ESN IN OLD ARRAYS
WAS ESN FOUND?
NO
YES
IS ESN TO BE DELETED?
NO
YES
DELETE ESN IN IGEN, IBKT, SHRINK ARRAYS, UPDATE POINTERS.
INSERT NEW EVENT AND ASSOCIATED DATA INTO IBKT AND IGEN
MERGE CURRENT ESN INPUT DATA WITH OLD IN IGEN AND IBKT ARRAYS UPDATE POINTERS
IS THIS CASE TO BE SAVED?
YES
NO
WRITE MERGED DATA ARRAYS TO DISC (IGEN & IBKT)
310

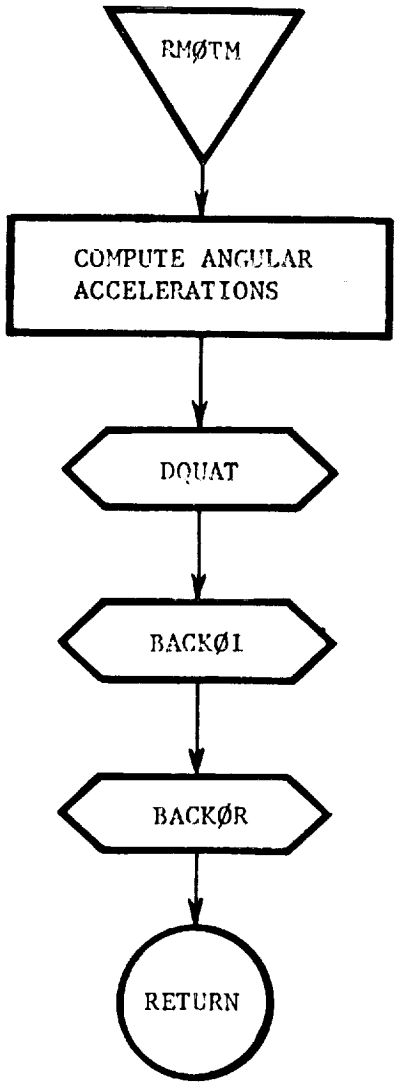
REVISE: This routine determines the indices of active constraints that are candidates to be dropped. This is done by expanding G1(I) in terms of the elements of the sensitivity matrix. The constraint with the most negative coefficient DFDC(I) is a candidate to be dropped.



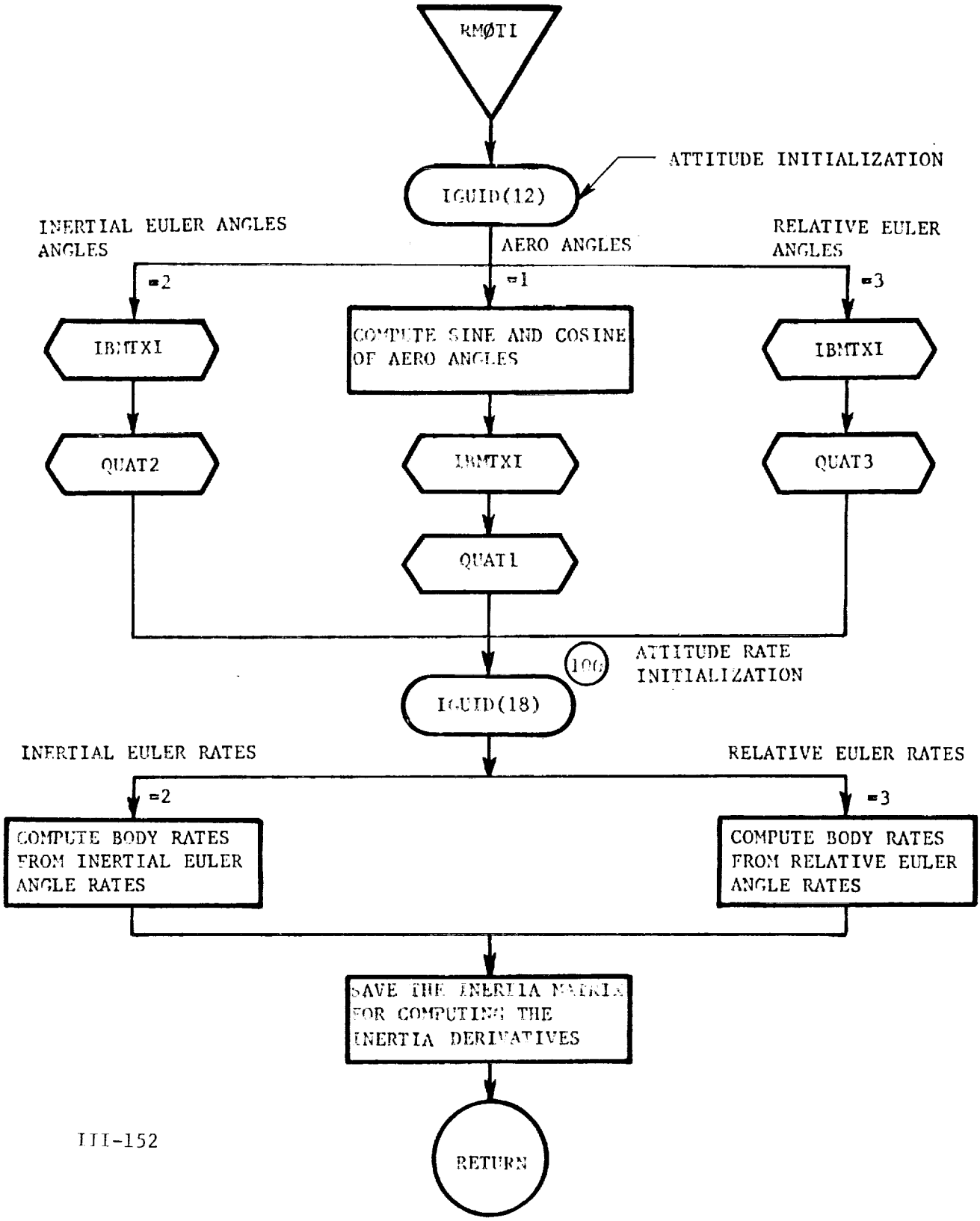
RGENDA: This routine reads namelist "GENDAT" and checks for any namelist errors.



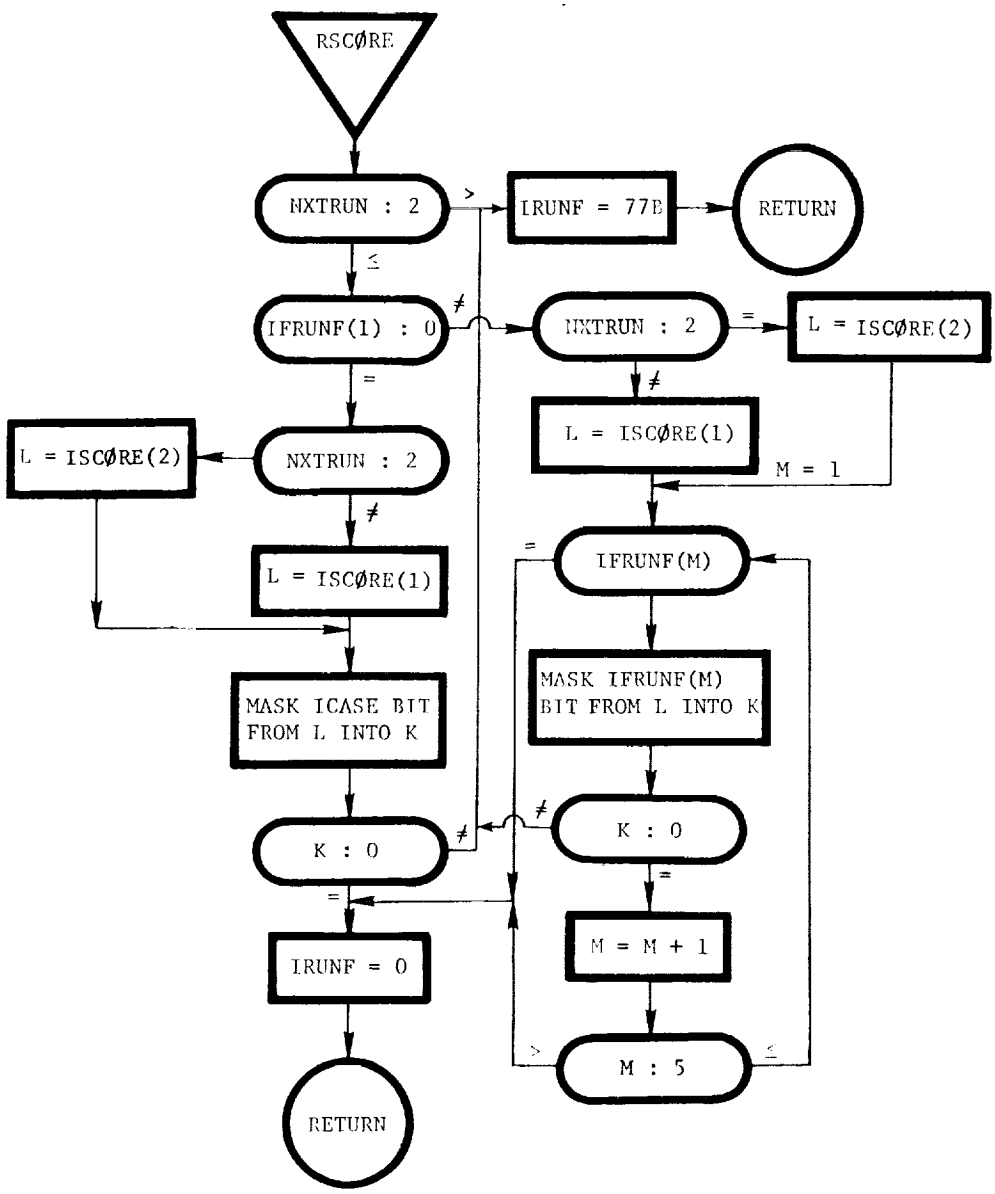
RMØTM: This routine calculates the rotational accelerations that result from the aerodynamic and the thrust moments.



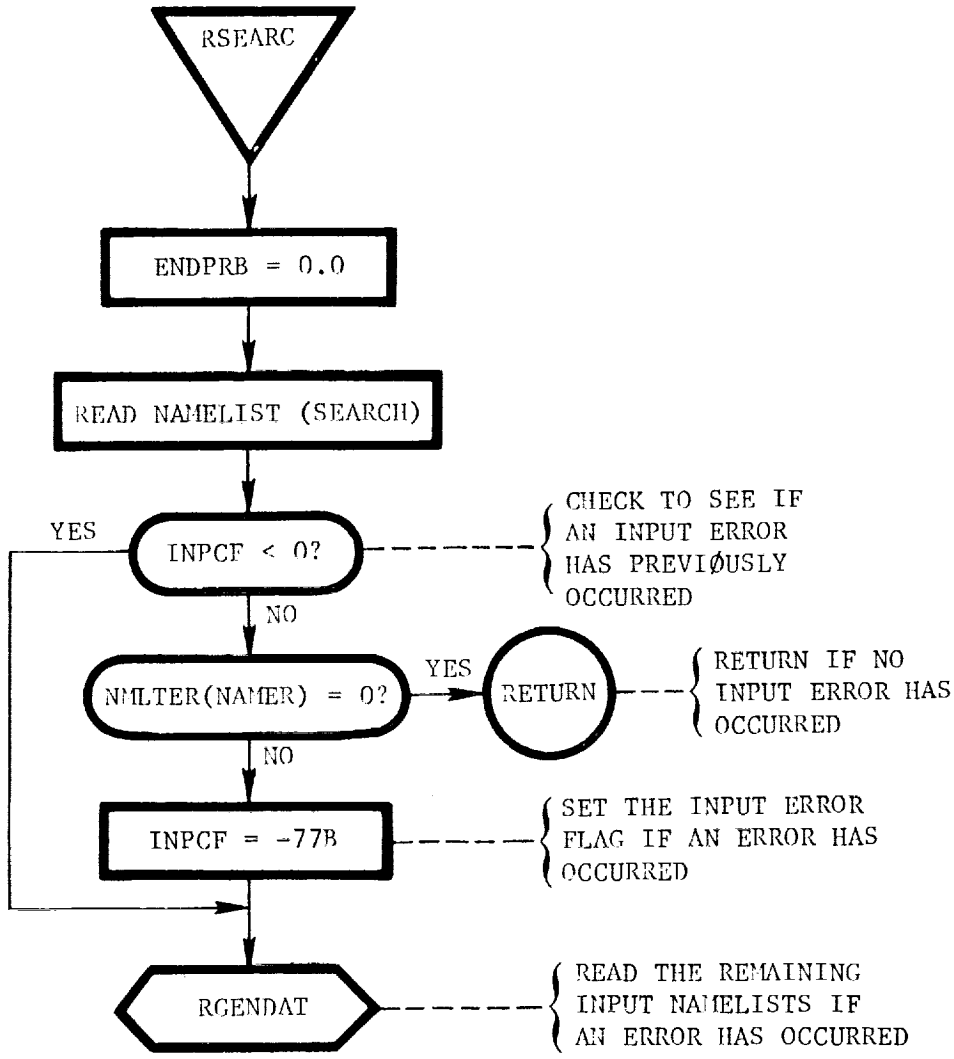
RMØTI : This routine initializes the quaternion parameters based upon the attitude input, and the body rates based upon the inertial or relative Euler angle rates.



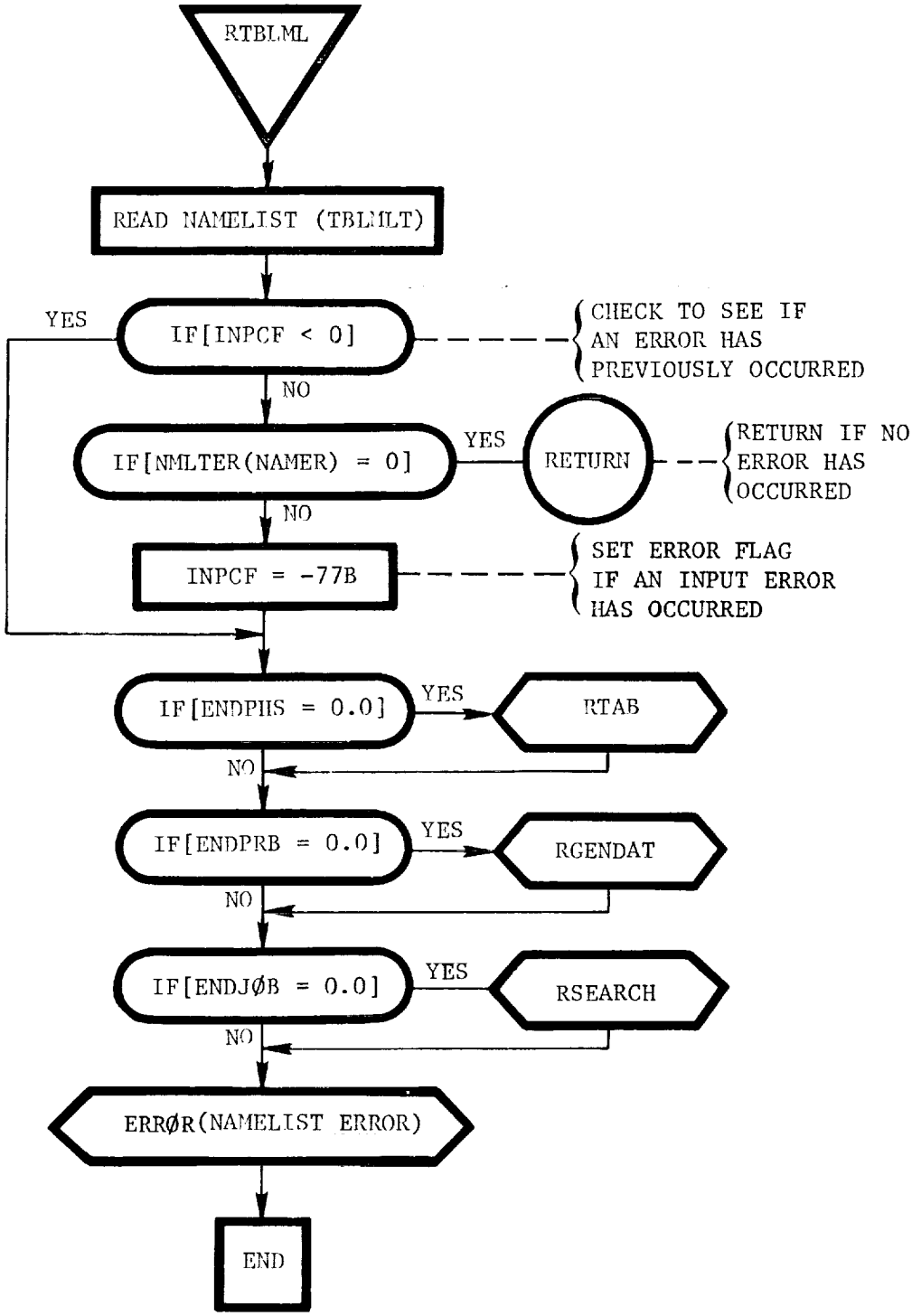
RSCØRE: This routine checks NXTRUN, IFRUNF(i), and ISCØRE to determine if criteria for executing the next case are met. A history of success or failure of previous cases are packed into ISCØRE by bit position. IRUNF is returned nonzero if the next case is to be executed.



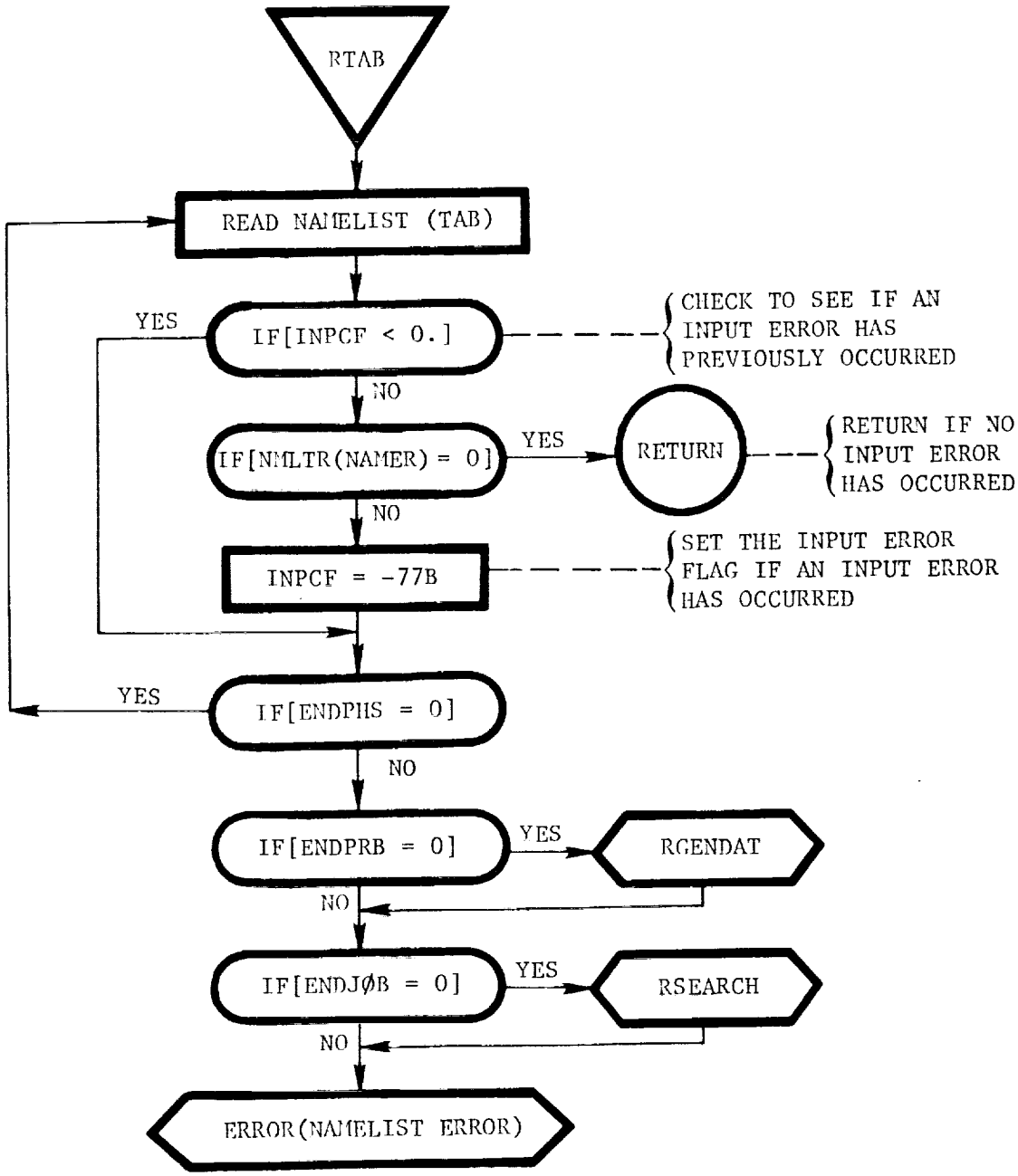
RSEARC: This routine reads namelist "SEARCH" and checks for any namelist errors.



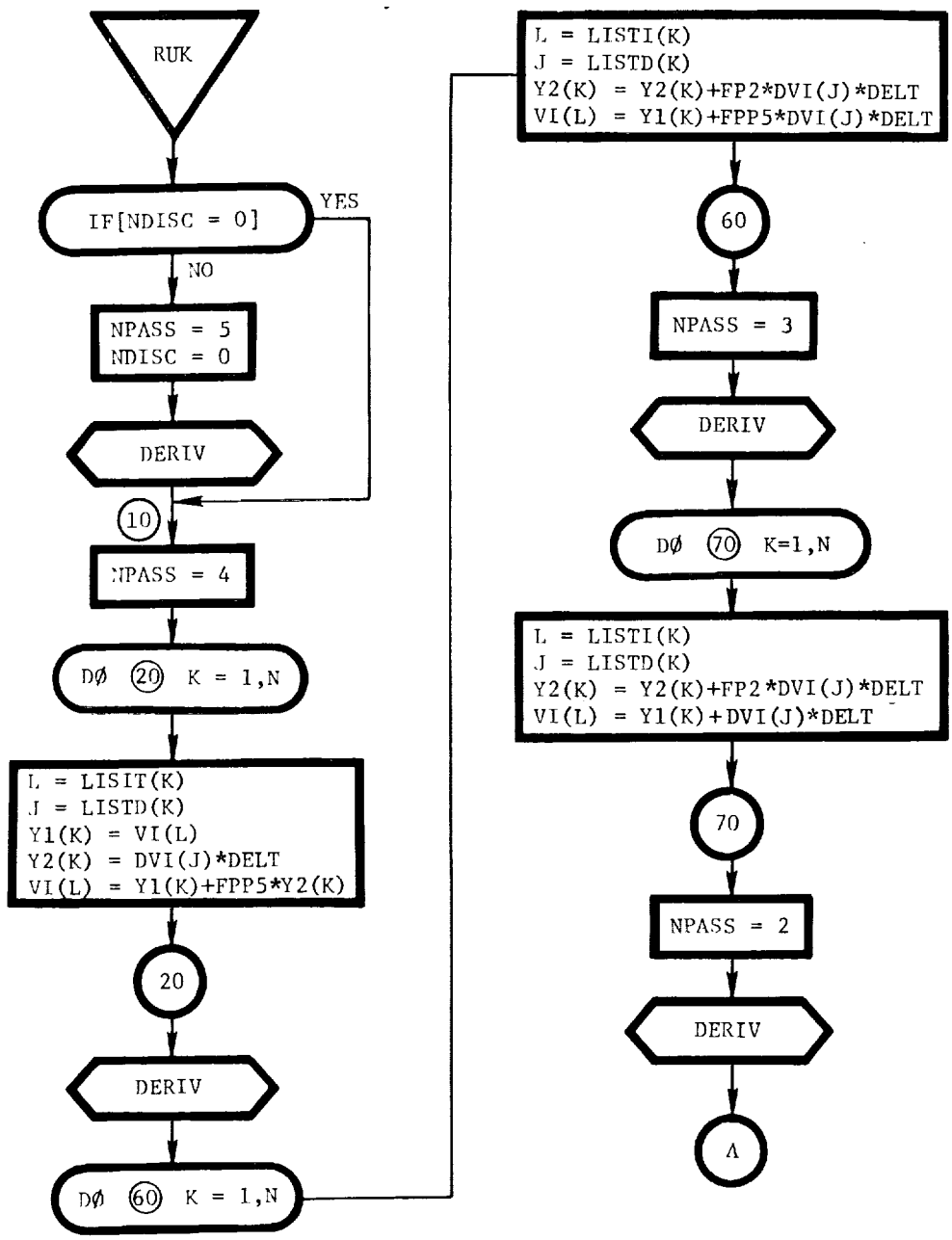
RTBLML: This routine reads namelist "TBLMLT" and checks for any namelist errors.

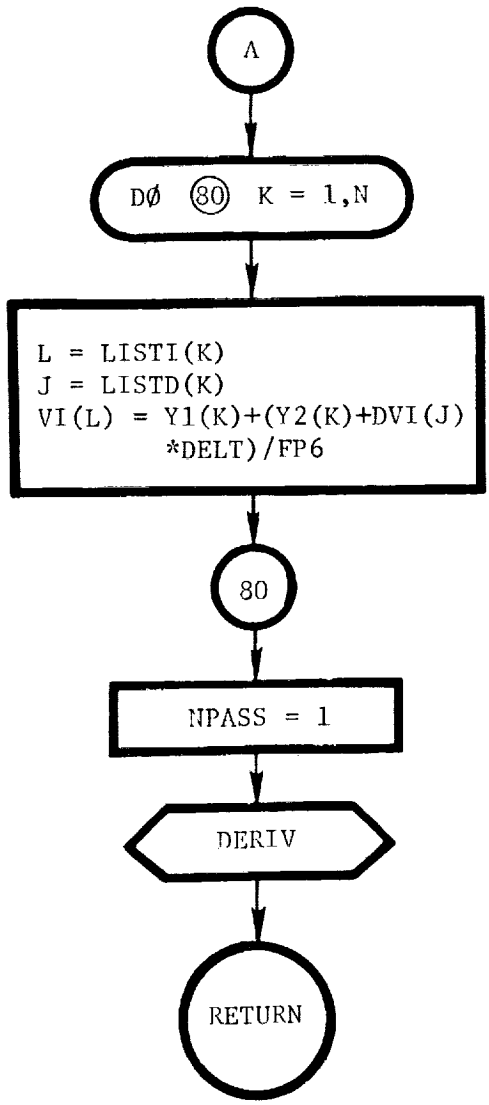


RTAB: This routine reads namelist "TAB" and checks for any namelist errors.

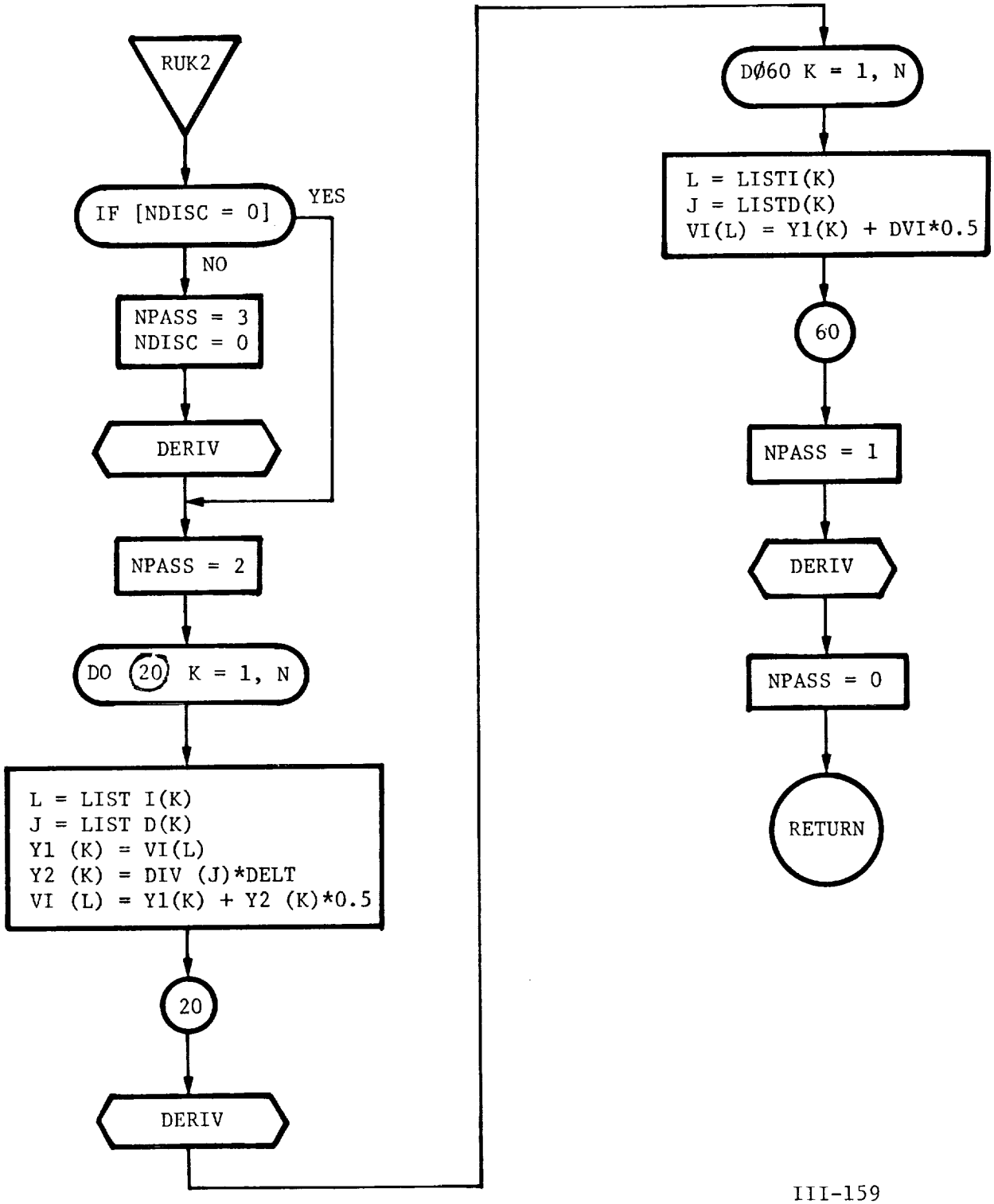


RUK: This routine contains the Runge-Kutta integration algorithm.

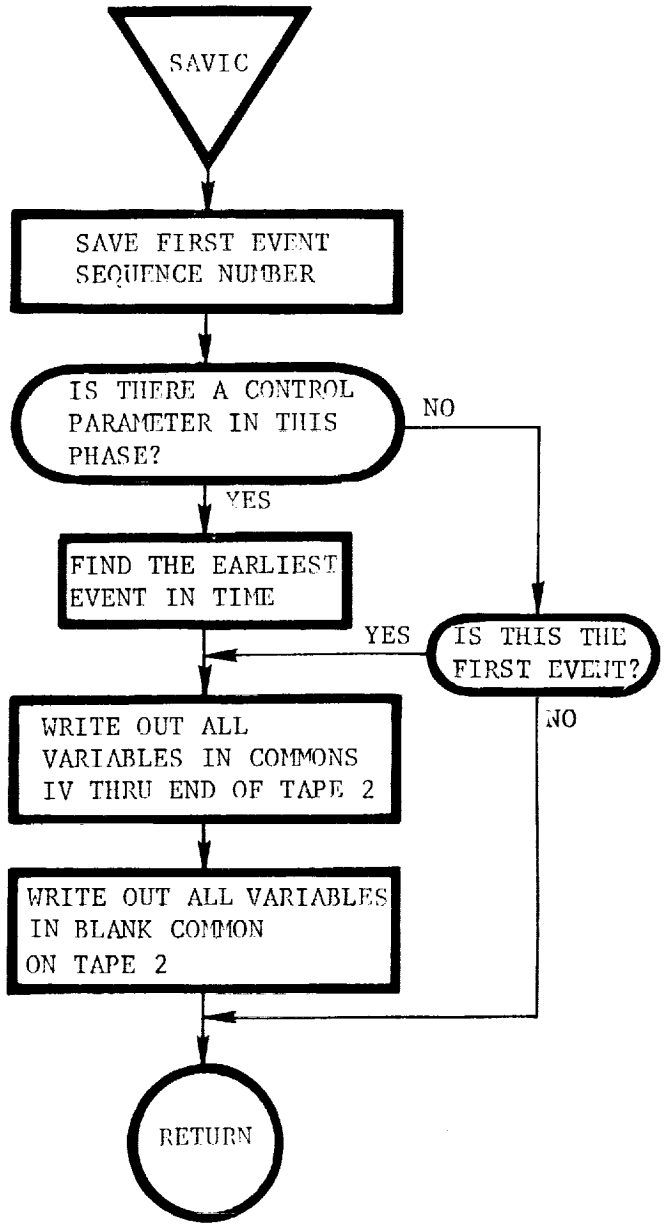



A
DØ 80 K = 1,N
L = LISTI(K)
J = LISTD(K)
VI(L) = Y1(K)+(Y2(K)+DVI(J)
*DELT)/FP6
80
NPASS = 1
DERIV
RETURN

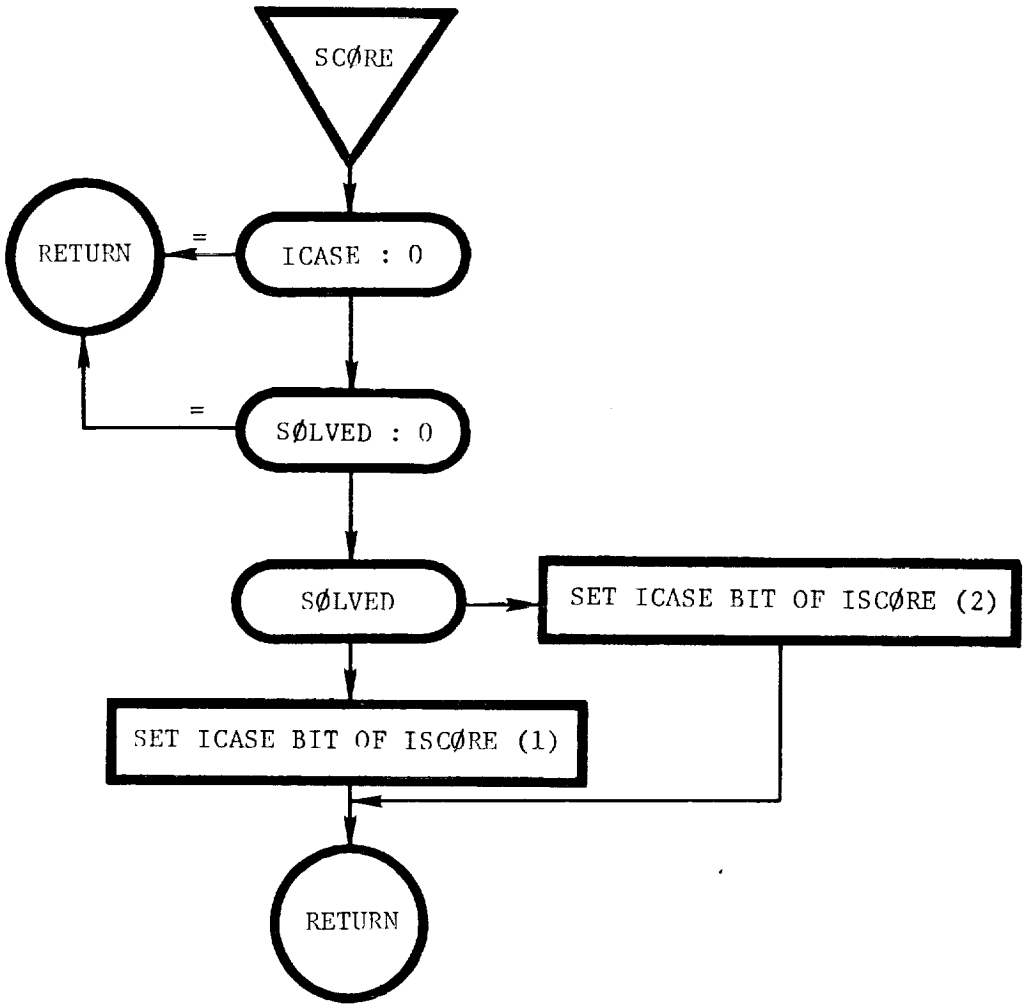
RUK2: This routine performs second order Runge-Kutta integration.



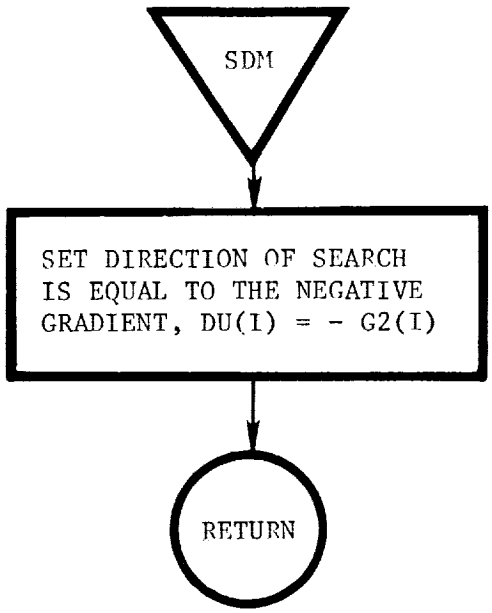
SAVIC: This routine buffers out commons IV and IBKT, which contain the state conditions, to file 2 at the beginning of each phase that contains an independent control variable. This information is used in running perturbed trajectories in the discrete-parameter mode.



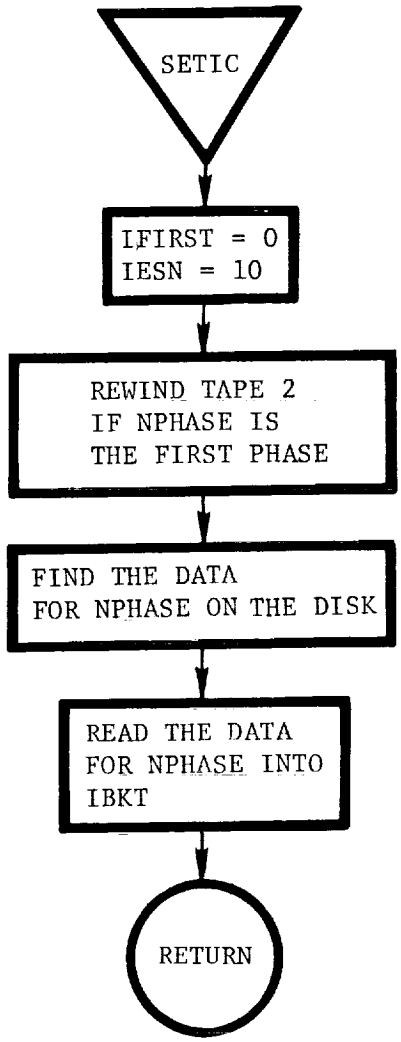
SCØRE: This routine sets the ICASE bit of ISCØRE(1) or ISCØRE(2) if SØLVED is + or -, respectively. No bit is set if SØLVED = 0.



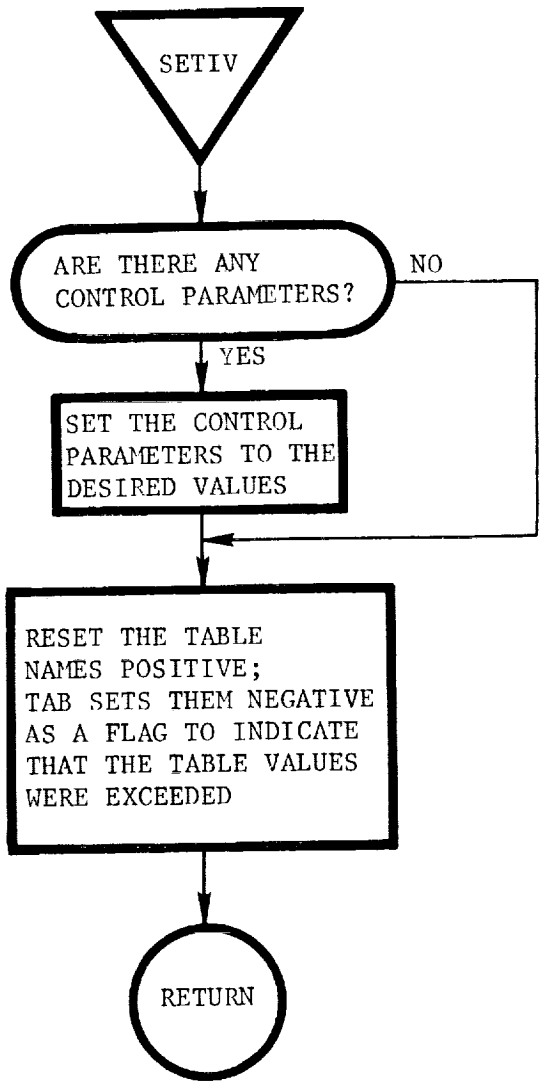
SDM: This routine computes the direction of search for minimizing P2 via the classic steepest-descent method.



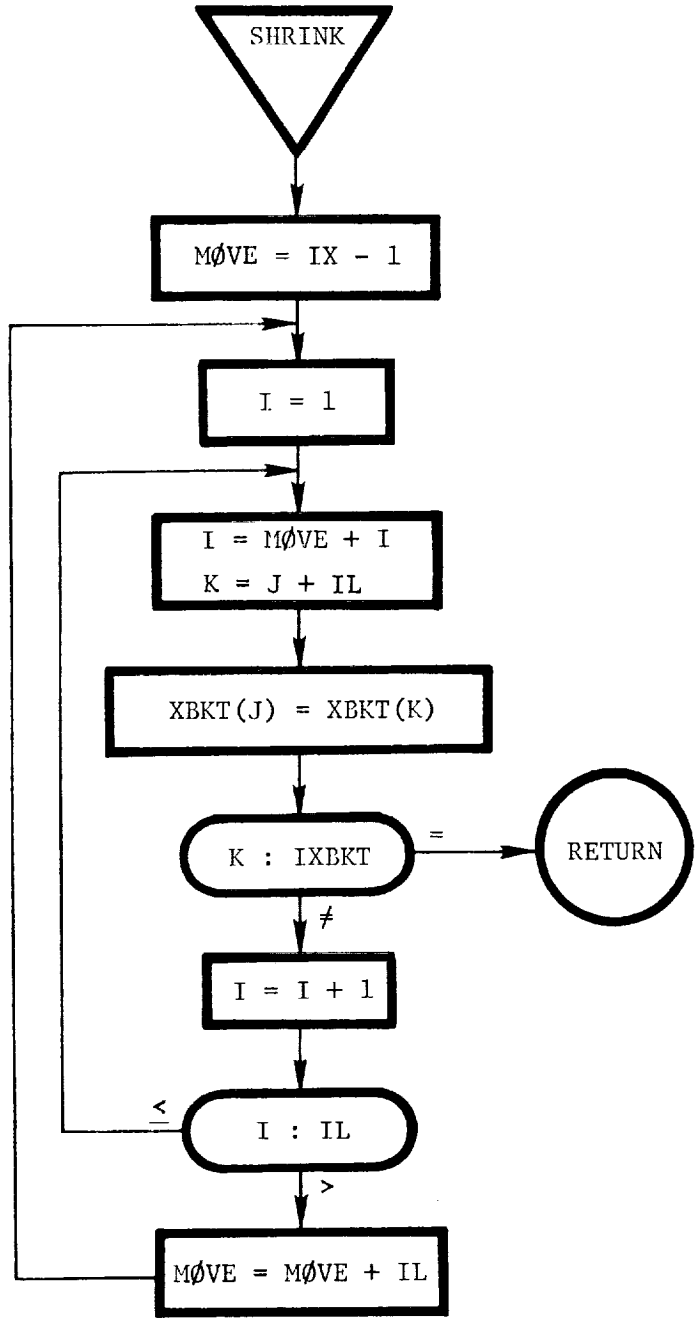
SETIC (NPHASE): This routine resets the initial conditions for the specified phase (NPHASE) equal to their nominal values.



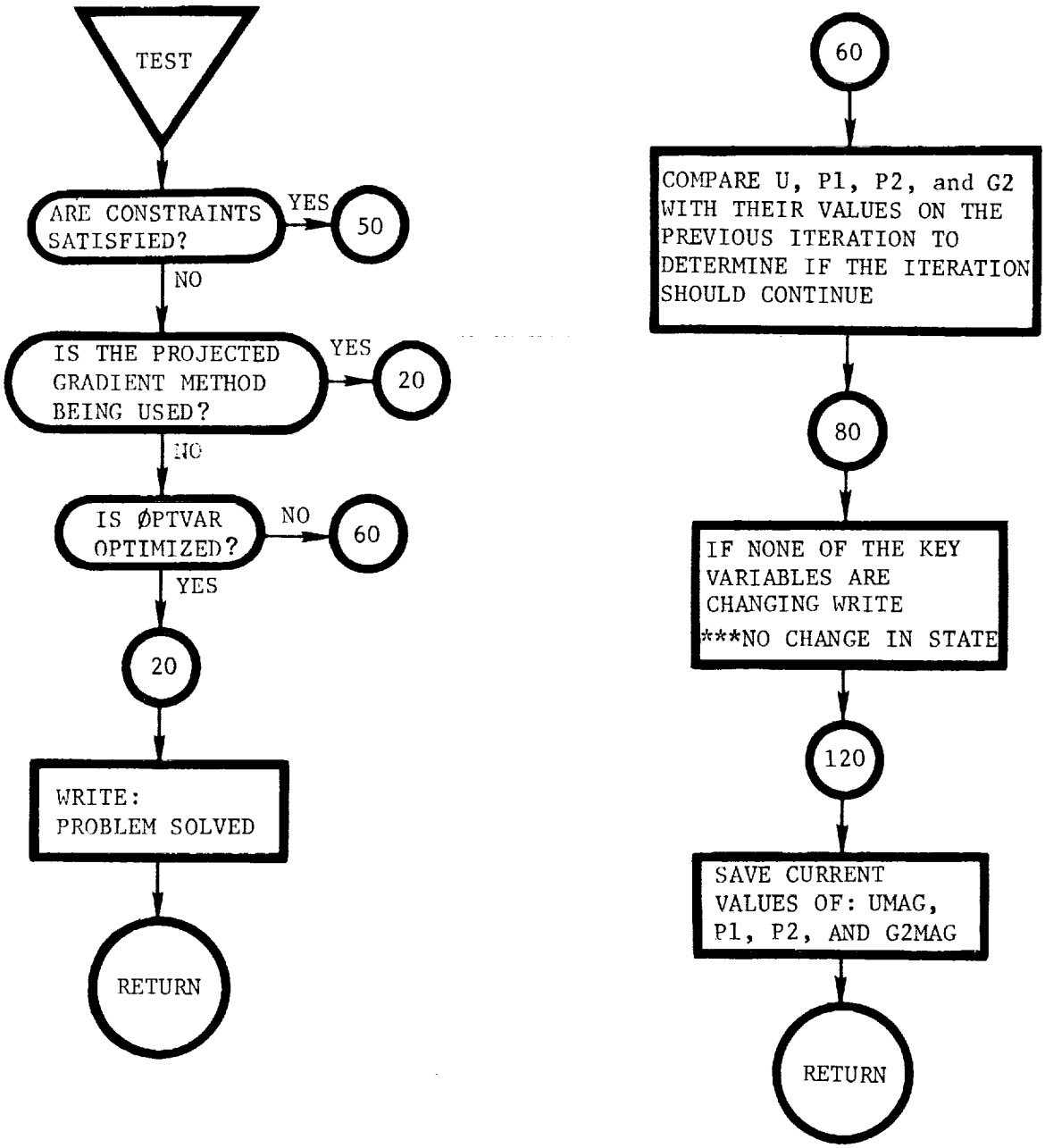
SETIV: This routine sets the control parameters to the desired values, based on the calculated control corrections.



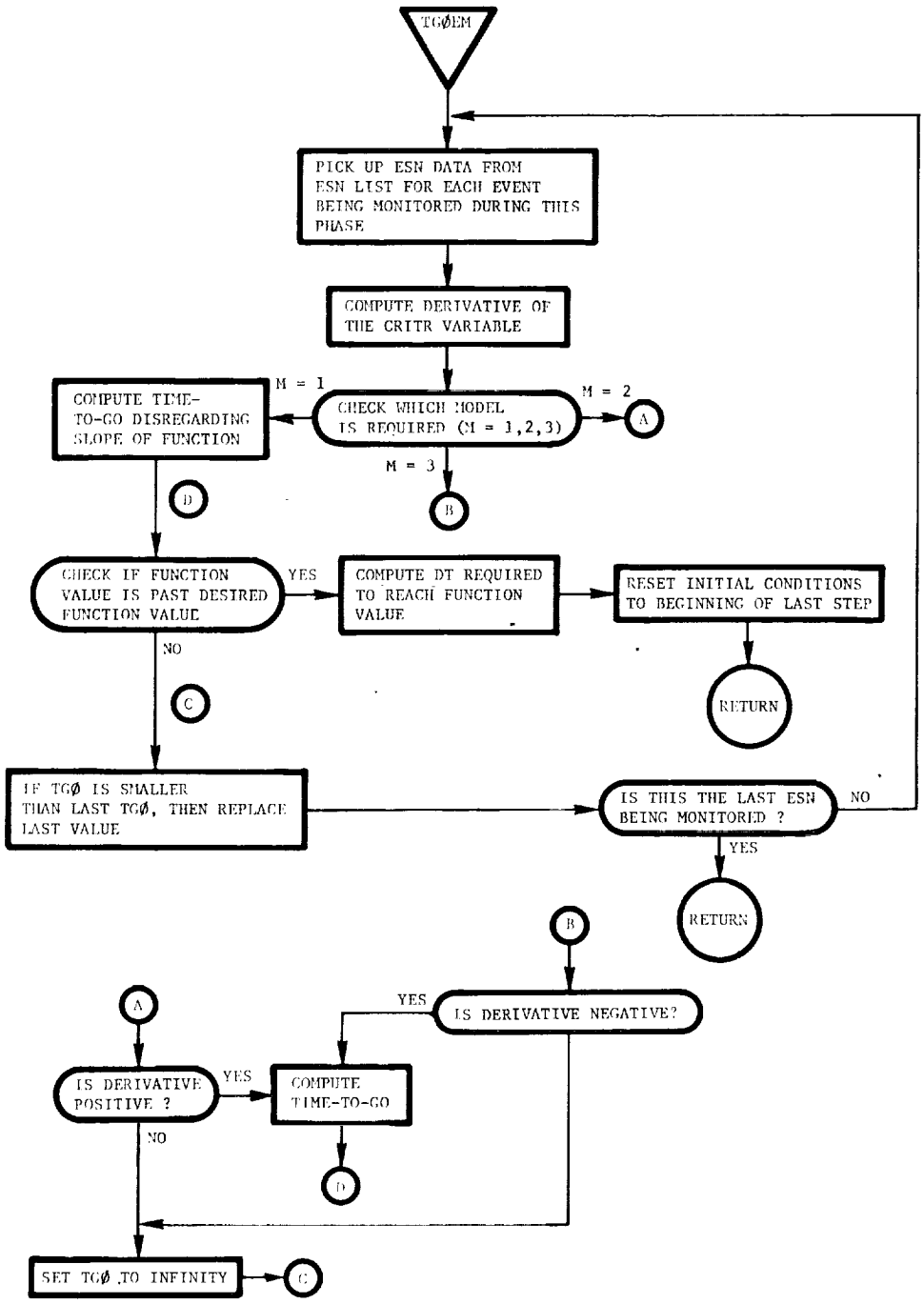
SHRINK (XBKT, IX, IL, IXBKT): This routine shrinks on array XBKT at position IX by IL words, where IXBKT is the total size of the array.



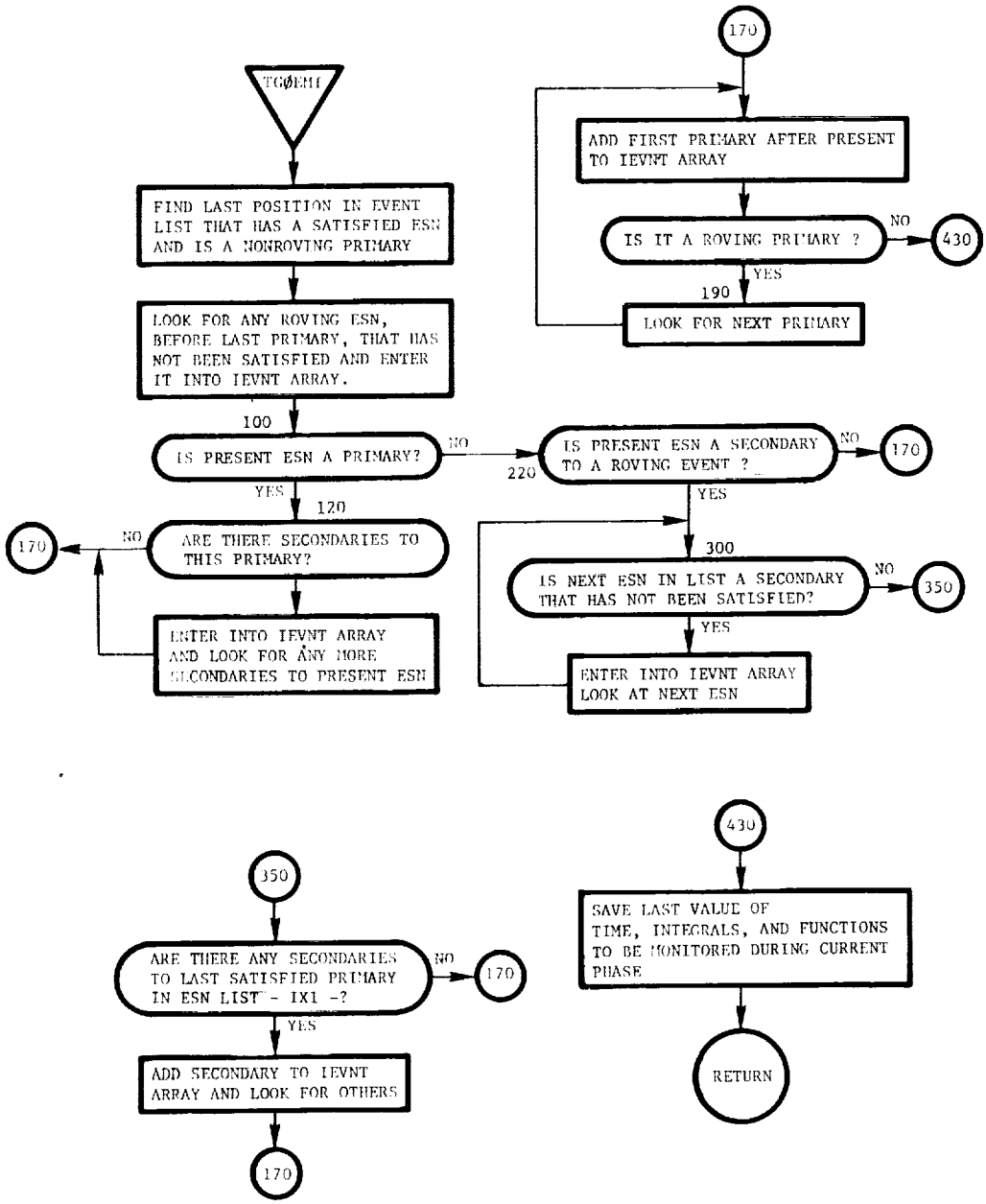
TEST: This routine tests for convergence.



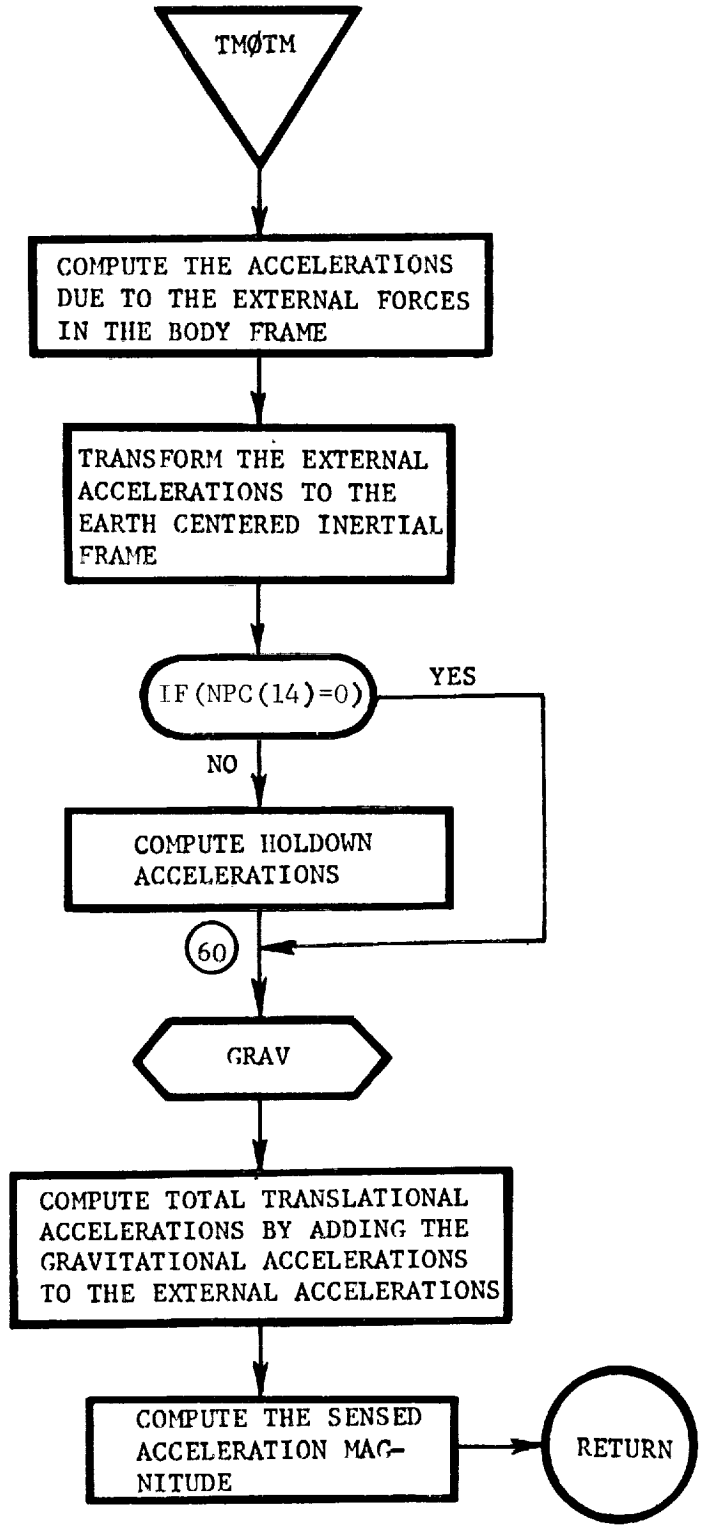
TGØEM: This routine computes the time-to-go with the next event for each criterion being monitored during the current phase. The smallest value is then selected and returned to CYCXM.



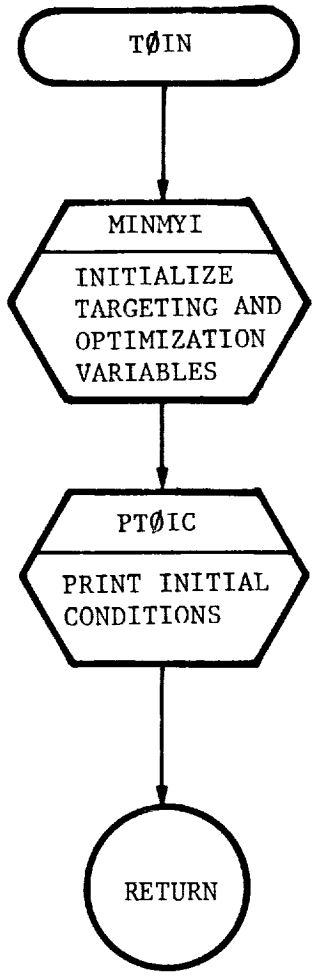
TGØEMI: This routine loads array IEVNT with the addresses of the events to be monitored during the current phase and specifies the order in which they are to be monitored.



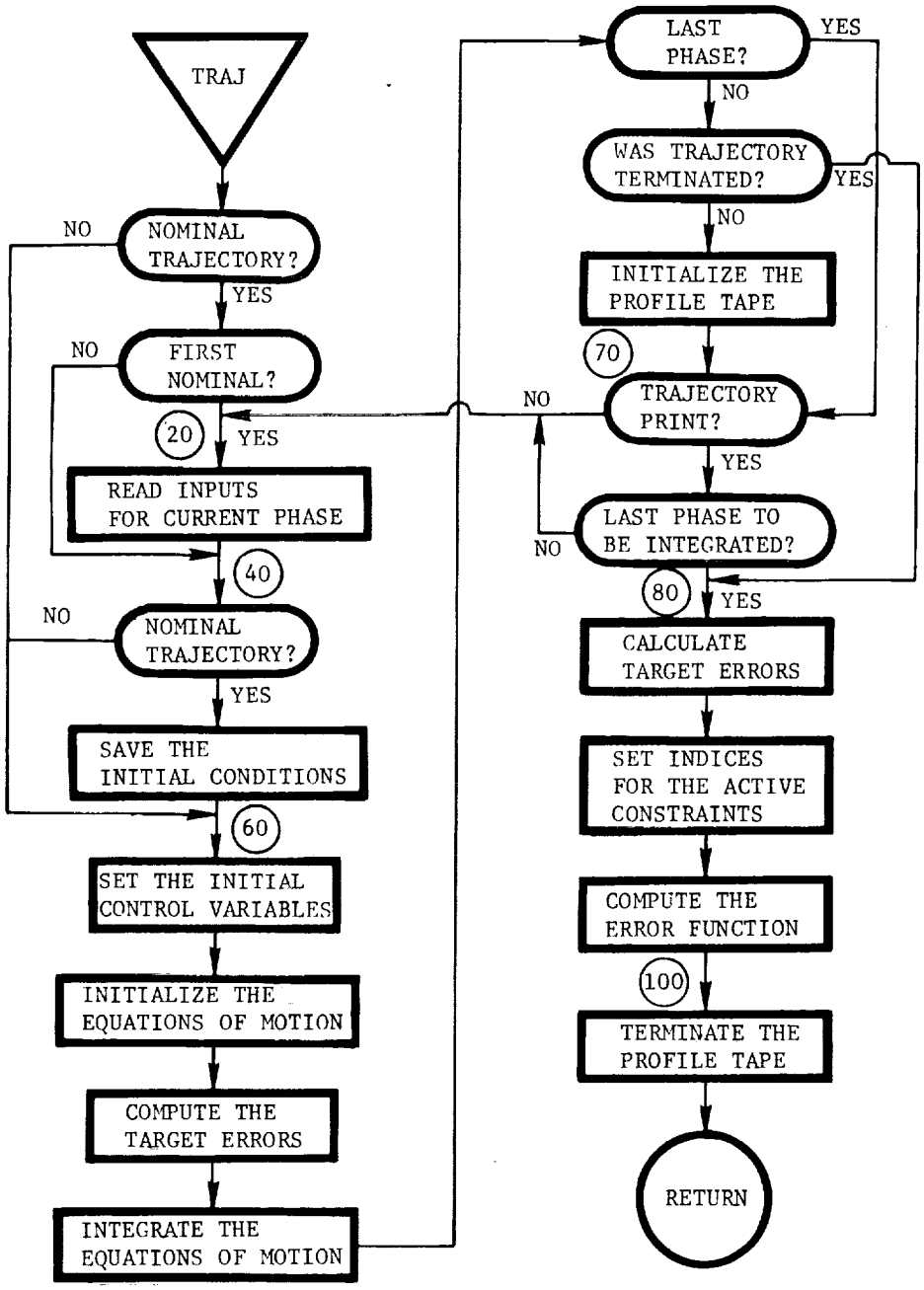
TMØTM: This routine calculates the translational accelerations of the vehicle in the Earth Centered Inertial frame.



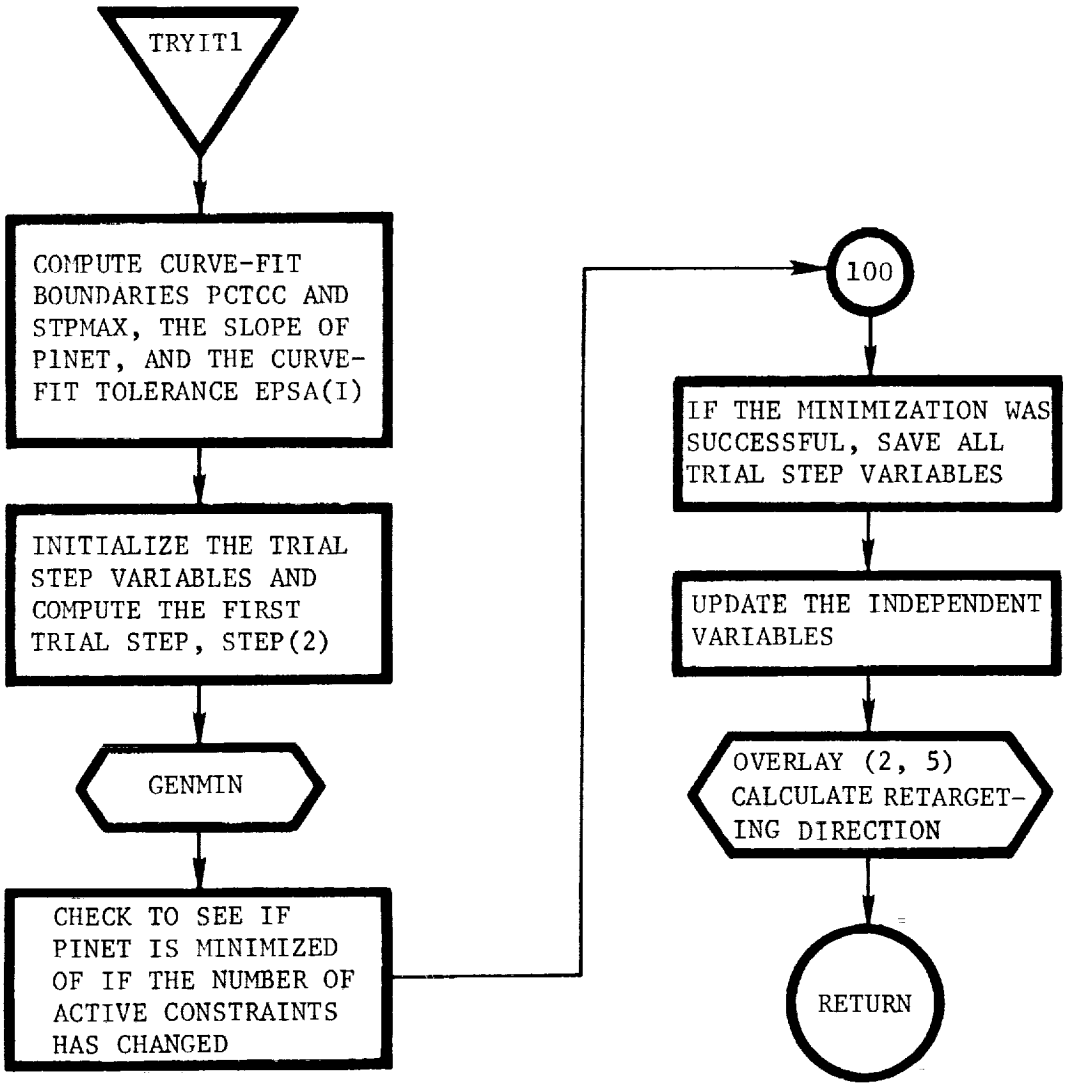
TØIN: Main program of overlay (2, 4). This routine calls MINMYI and PTOIC.



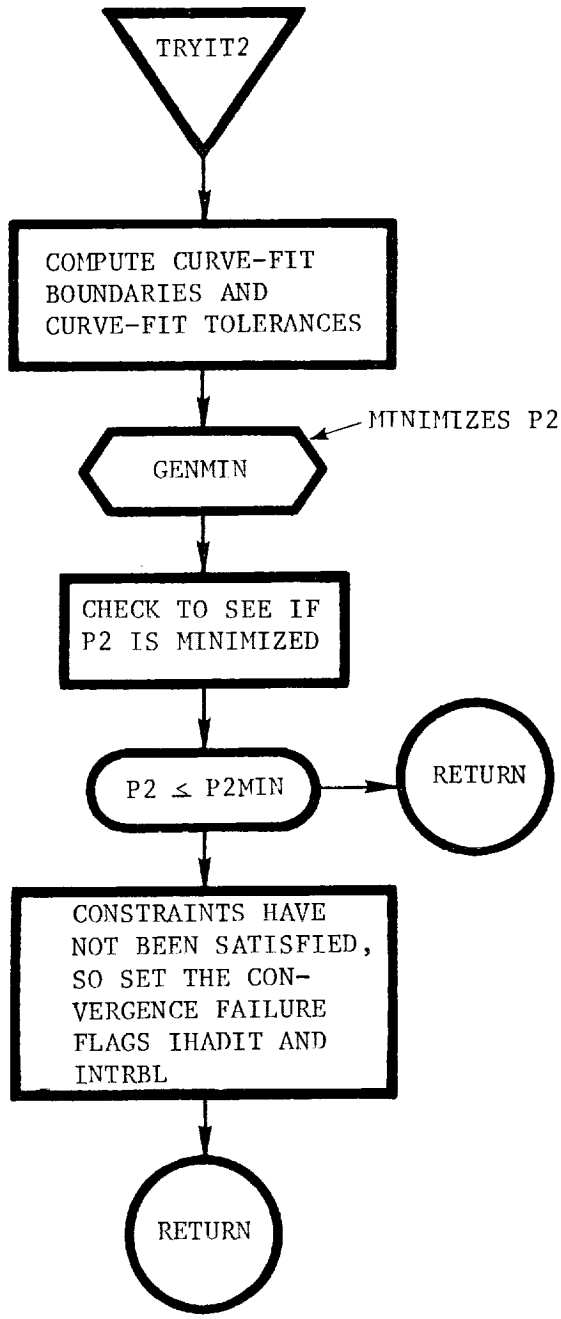
TRAJ: This routine propagates a trajectory from the beginning of the specified phase to the final cutoff condition.



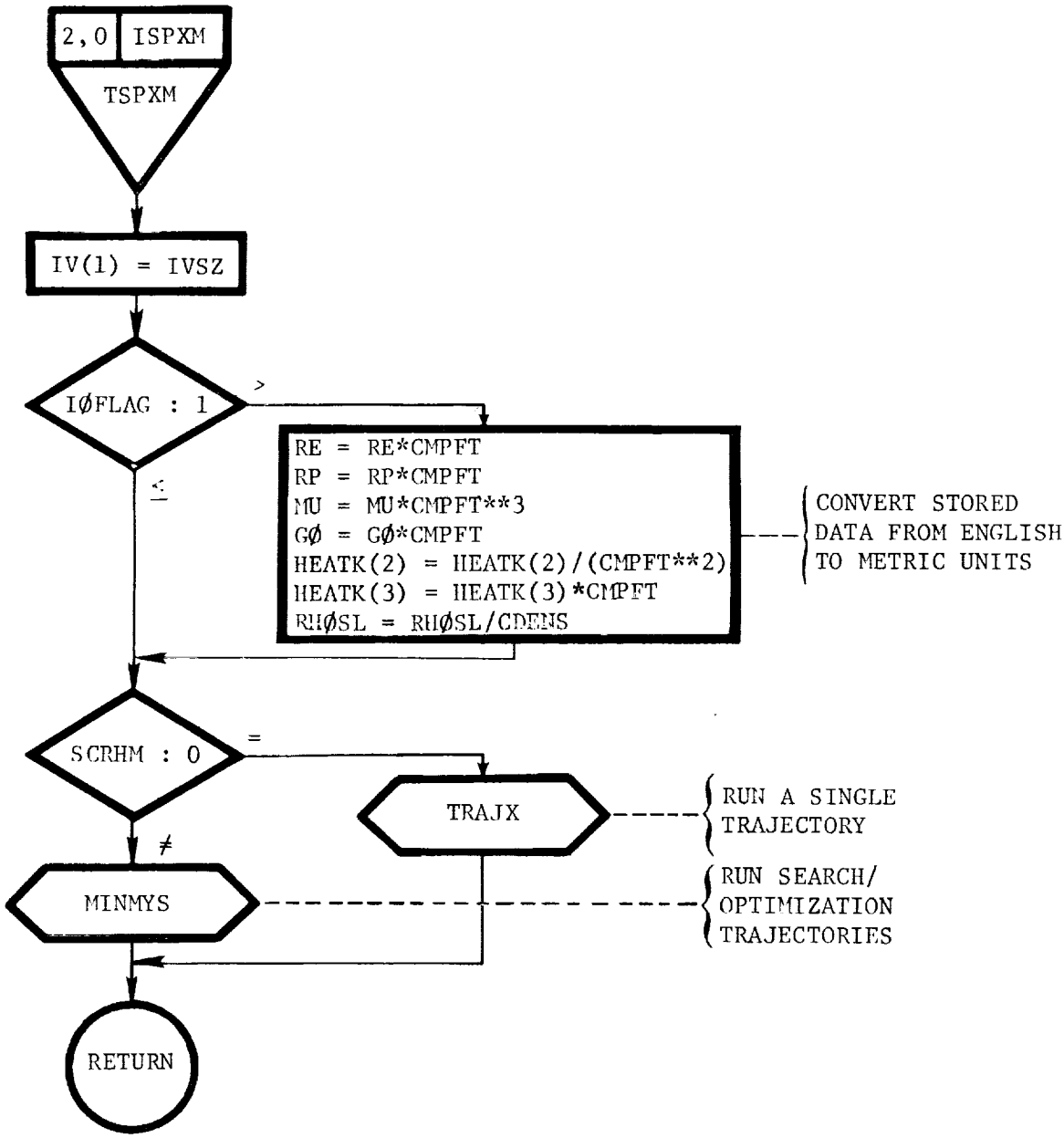
TRYIT1: This routine minimizes the estimated net cost function as a function of the step-size parameter. The principal function of the routine is to setup of the data required by GEMNIN, where GENMIN is the routine that actually minimizes the function.



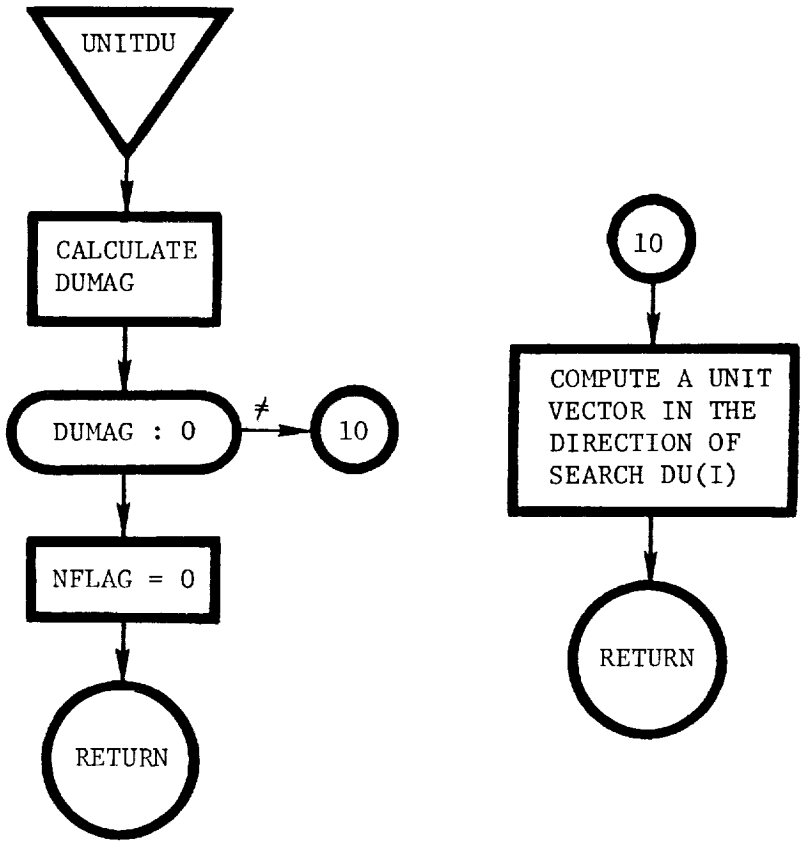
TRYIT2: This function minimizes the constraint error function P2.



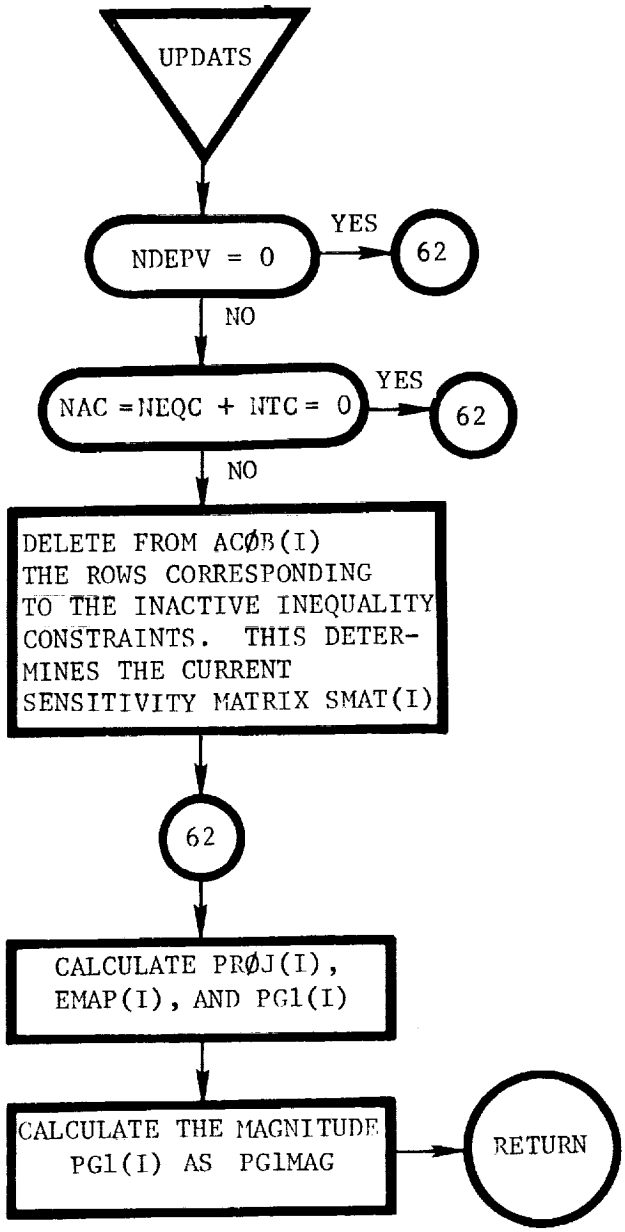
TSPXM: This routine is the main program of overlay (2, 0) and controls the overall operation of the trajectory simulation routines.



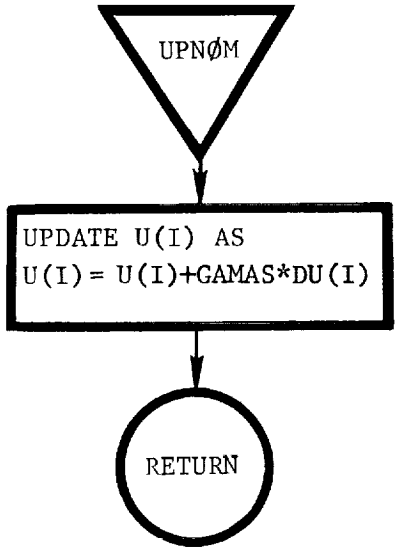
UNITDU: This routine unitizes the control correction vector and computes the magnitude of the control correction vector.



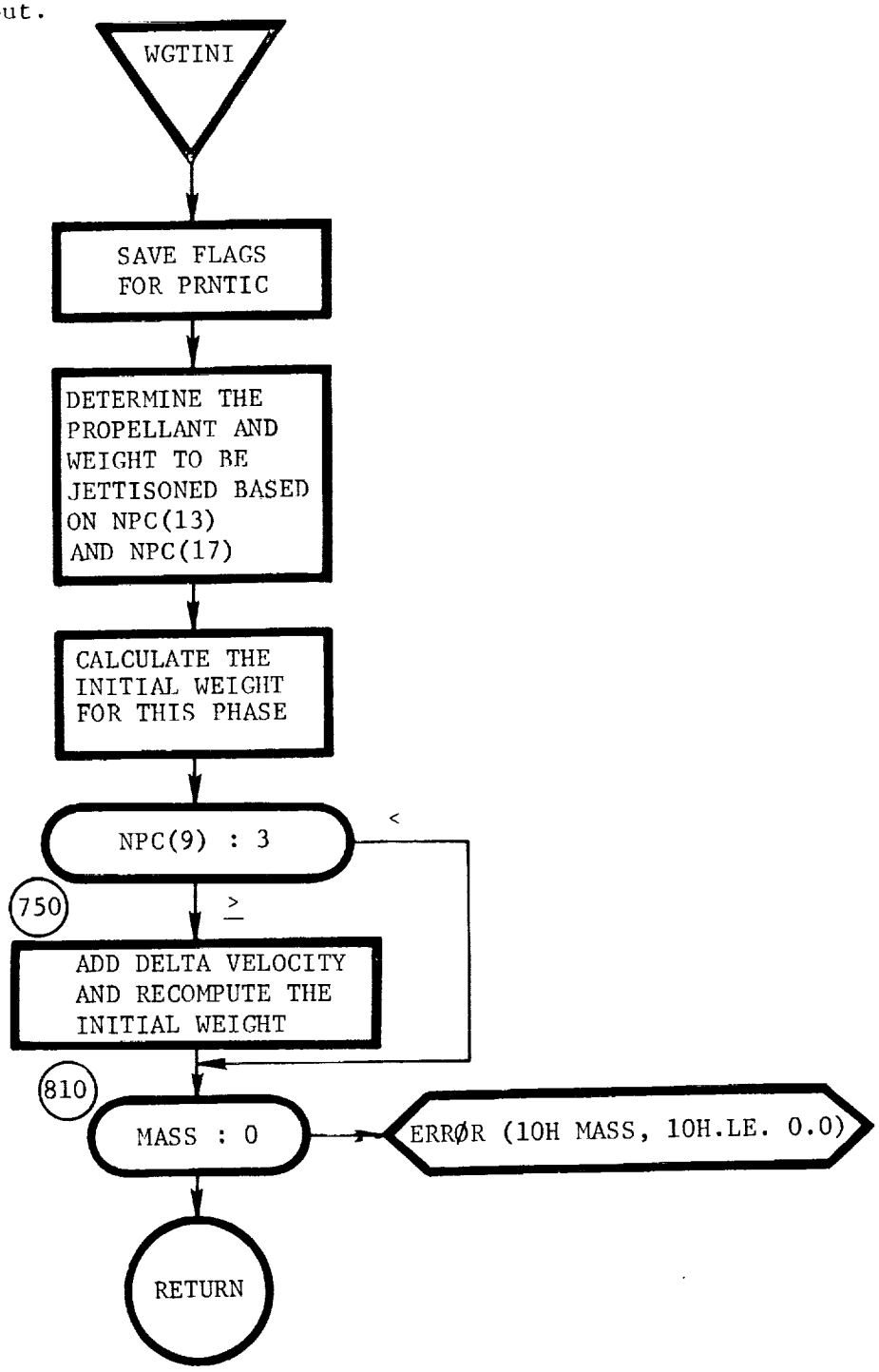
UPDATS: This routine updates the sensitivity matrix by deleting those rows that can be dropped. The routine also computes the projection matrix PRØJ(I) and the error map matrix in subroutine PGM to determine the direction of search.



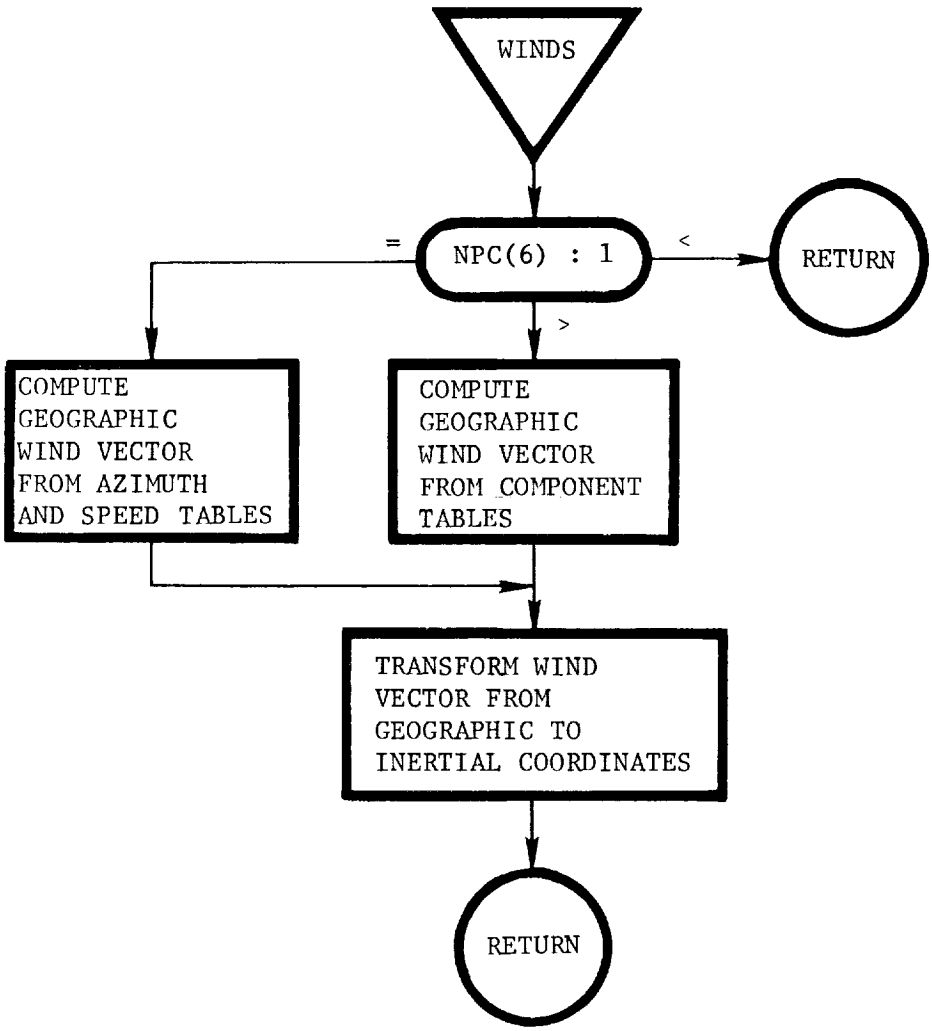
UPNØM: This routine updates the nominal values of the independent variables according the direction of search DU(I) and the step size GAMAS.



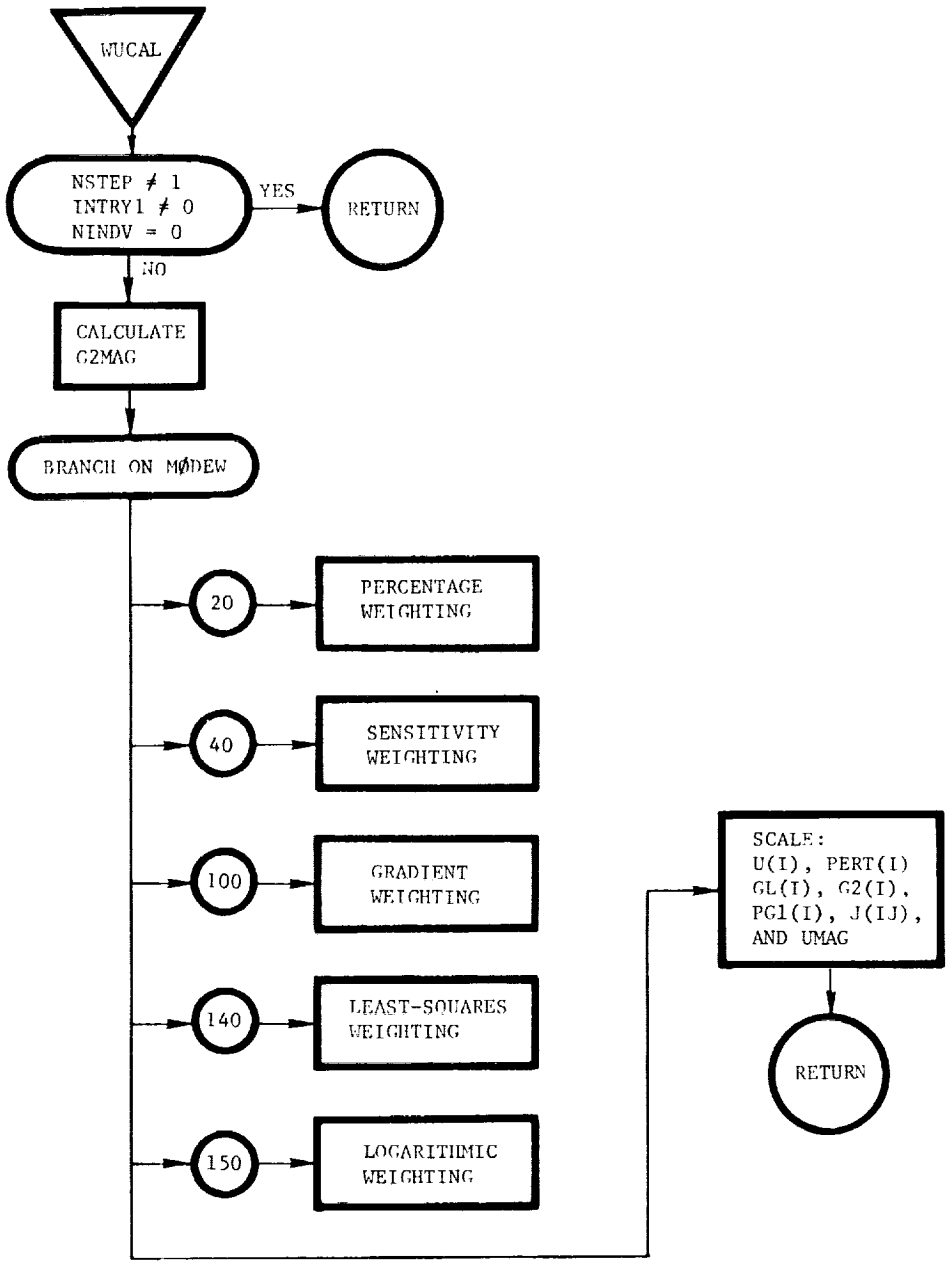
WGTINI: This routine initializes the vehicle's weight for the current phase and adds the specified delta velocity if requested by user input.



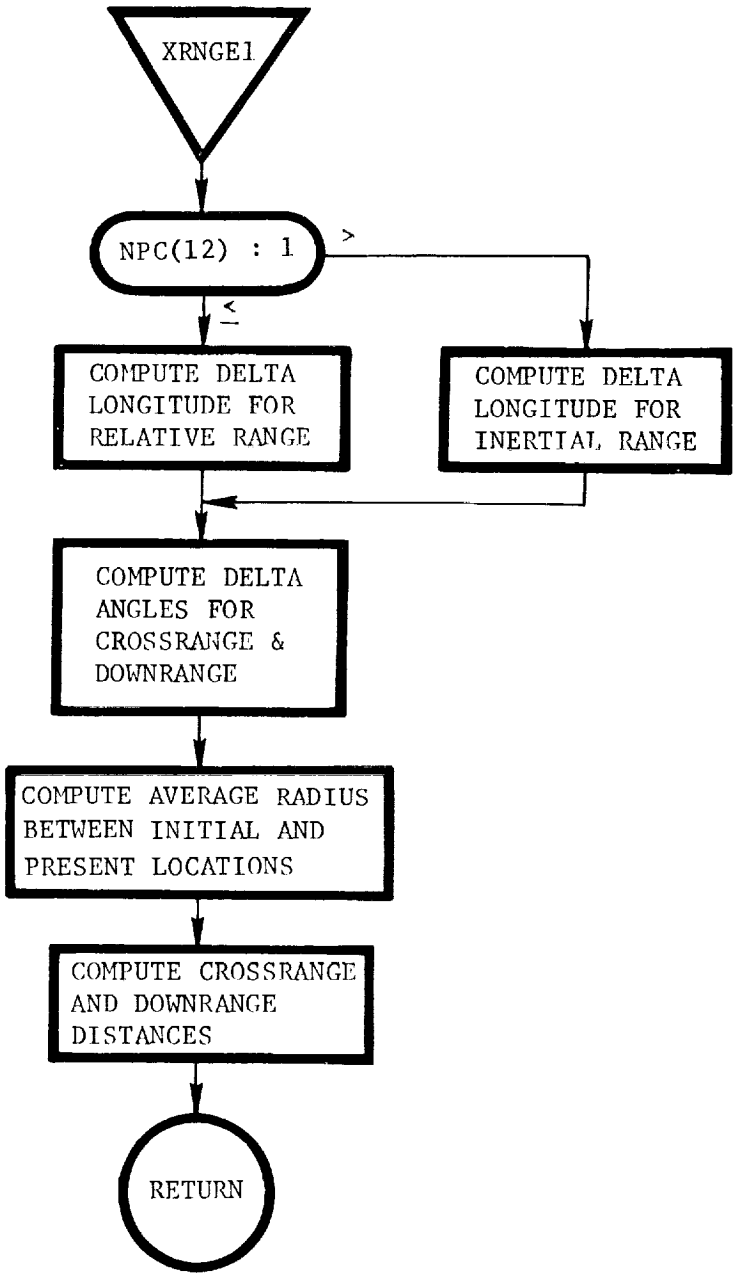
WINDS: This routine calculates the components of the wind velocity vector in the Earth-centered inertial coordinate system.



WUCAL: This routine calculates the weighting matrix for the independent variables.



XRNGE1: This routine calculates the downrange and crossrange distances, based on relative great circles.



IV. SERVICE ROUTINES

The following service routines are used to perform arithmetic tasks that are used frequently throughout the program.

ABT (A,B,C,L,M,N): This routine computes matrix C given matrices A and B, the dimensions of A (L by M), and the dimensions of B (M by N) as matrix A times the transpose of matrix B.

ABXT: This dummy routine is used to force the loading into the memory region of overlay (0,0) several system routines.

ANGLE2 (X,Y,ALPHA): This routine computes the angle ALPHA, in degrees, between vectors X and Y. ALPHA is measured counterclockwise from X to Y and can range from 0 to 360 deg.

ANGV2 (A,B,REF): This function computes the angle from unit vector A to unit vector B about the reference vector REF, where A,B and REF are unit vectors.

ATANH (A): This function returns as the hyperbolic arc-tangent of A.

ATAN3 (A,B): This function computes the angle, from 0 to 2 pi radians, whose arc-tangent is A over B.

BTW (B,W,D,L,M,N): This routine computes matrix D as the transpose of matrix B times matrix W, where L, M, and N are the dimensions of matrices B and W.

BUCKET (X,Y,N,XX,YY,NP): This routine rearranges an array X and the corresponding elements of Y in ascending order. N is the number of elements in each array. NP is a pointer indicating the first element of Y that is less than the next element. NP is zero if it does not exist. The ordered arrays are returned as XX and YY.

CONVO1: This routine integrates first order linear differential equations using the convolution integration technique.

CUBMIN (A,XMIN,YMIN): This routine evaluates a cubic polynomial coefficients (A), and returns the minimum values of the function (YMIN) and the corresponding value of the argument (XMIN).

DIGDIF (M,N,NDIF): This routine finds the number of different digits (NDIF) between M and N, where M and N are base-10 numbers.

EPHEM (DATE, GULIE, GHA, RAS, DECL): This routine computes the Greenwich hour angle, and the right ascension and declination of the sun given the month/day/year date and the floating point julian date from 1950.0.

ERRØR (I,J): This routine writes an error message and determines if it is fatal or nonfatal. I and J are Hollerith words. If the first letter in I or J is blank, the error is fatal; otherwise it is nonfatal.

EXPN(IBKT,IX,IK,IXBKT): This routine moves the contents of IBKT from IXBKT down IL locations, and shifts the contents of IBKT from IX down IL locations. IXBKT is the last occupied cell in IBKT.

EXPN (IBKT,IX,IL,IXBKT): This routine moves the contents of IBKT from IXBKT down IL locations, and shifts the contents of IBKT from IX down IL locations. IXBKT is the last occupied cell in IBKT.

FØPMIN (X,U,XMIN,YMIN,IERR): This routine calculates the minimum of a polynominal based on four ordered pairs (X,U). The abscissa value that minimizes this cubic polynomial is returned as XMIN and the corresponding ordinate value is returned as YMIN.

FORMIN (CENJUL, GULIE1, RANUT): This routine computes the duration of the right ascension of the Sun given the true Julian Century and the Julian date.

GENTAB (TABLE): This function is the general table interpolation routine, where TABLE is the location of the table in IBKT. This routine replaces the functions TAB, BTAB and TRITAB used in previous versions of POST6D.

INVM(A,N): This routine inverts an N by N matrix A and returns A as the inverse.

LDRL: Service module load relief filter employed by the general analog autopilot.

MATPY (A,B,C,L,M,N): This routine multiplies matrix A by matrix B to produce matrix C, where A is an L by M matrix and B is a M by N matrix. Thus, C is an L by N matrix.

MOTAB (XS,X,N1,N2,LIMIT): This function performs the bracketing of the independent argument XS. X is the first independent value of the table, N1 is the increment to the next largest independent value of table, N2 is the location of the last point used from table, and limit is the last point in table.

MTRXM (A,B,C): This routine multiplies matrix A by the transpose of matrix B to form matrix C, where A, B, and C are 3 by 3 matrices.

MTRXTV (AT,V,W): This routine multiplies the transpose of matrix A by vector V to form vector W, where A is a 3 by 3 matrix and V and W are 3 by 1 vectors.

MTRXV (A,B,C): This routine multiplies a 3 by 3 matrix A by a vector B, and returns the answer as C.

PAD (A,B,MØD): This routine determines the delta X that produces the most precise derivative without losing significance in being rounded off.

Argument	Calls	Call
	1 to (n - 1)	n
A	f(X)	X
B	f(X + X)	Dummy
MOD	0	1

PØLY (N1,CØFI,ARGI): This function evaluates a polynomial of degree N1 with coefficients CØFI as a function of ARGI.

PPT (P,C,M,N,S): This routine computes matrix C as matrix P times the transpose of matrix P, where P is an M by N matrix and S is a scalar that must be set to zero in the calling program.

PQ1: This routine calculates the necessary constants for the convolution integration technique.

QUADMN (A,XMIN,YMIN,MINFLG): This routine determines the minimum point of a quadratic polynomial, where A(K) is the coefficient of the (K-1)th degree term in the quadratic polynomial, XMEN is the minimum Abscissa value, YMIN is the minimum ordinate value and MINFLAG is set nonzero when a minimum does not exist.

SERCH (I,IV): This routine locates the address of variable I (Hollerith) with respect to the beginning of the common reference.

SP (X,Y,N): This function computes the scalar product of two N-dimensional vectors X and Y.

SREL (LIST): This routine locates the addresses of the variables in the array LIST, where LIST is constructed as follows:

NAME1,NAME2,X_1,
NAME3,NAME4,X_2,
NAME5,NAME6,X_3,

The routine is also used to find the addresses of the derivatives and their corresponding integrals.

SUN (AJP, ARA, DEC): This routine computes the right ascension and declination of the Sun.

SYMATS (A,S,B,M,N): This routine multiplies matrix A times the upper triangular symmetric matrix S and stores the result as matrix B. All matrices are stored by columns, and the dimensions are MXN.

SYMSTA (S,A,B,M,N): This routine multiplies the upper triangular symmetric matrix S times matrix A and stores the result as matrix B. All matrices are stored by columns, where the matrix dimensions are MXN.

THPM (X,Y,XMIN,YMIN): This routine fits a quadratic polynomial through three points. It returns the minimum of this polynomial as YMIN and the minimizing value of X as XMIN.

THPØSM (X,Y,DYDX1,XMIN,YMIN): This routine fits a cubic polynomial using three points and the slope of the function. The routine returns the minimum of the polynomial as YMIN and the minimizing value of X as XMIN.

TPØSM (X,Y,DYDX1,XMIN,YMIN): This routine fits a quadratic polynomial using two points and the slope of the function. It returns the minimum of the function as YMIN and the minimizing value of X as XMIN.

D	=	the array containing the table to be interpolated.
N(1)	=	number of x's
N(2)	=	number of y's
N(3)	=	number of z's
N(4)	=	type of interpolation
N(5)	=	type of x's
N(6)	=	type of y's
N(7)	=	type of y's
N(8) thru (20)	=	pointers for the last used point on the table.
X*,Y*,Z*	=	values of the table arguments

VCRØSS (A,B,C): This routine computes the vector cross-product, C, between two three-dimensional vectors A and B.

VDØT (X,Y): This function computes the dot product between two three-dimensional vectors X and Y.

VUNIT (X,XBAR,XMAG): This routine computes the magnitude, XMAG, and unit components XVAR, of a 3-dimensional vector X.

ZWASHF (EOUTH, EIN, EINH, TAU): This routine computes the $S/S+\tau$ filter using the Z-transform, assuming a triangular hold.

ZEROM (A,M,N): This routine zeroes the M by N vector A.

ZLAG (EOUTH, EIN, EINH, TAU): This routine computes the $\frac{1}{S+\tau}$ filter using the Z-transform, assuming a triangular hold.

V. DEFINITION OF INTERNAL FORTRAN SYMBOLS

This section presents the symbols used internally in the program. Most variables in the program are located in COMMON to conserve core locations. Certain variables are local variables to a specific routine. These types of variables are not shown in this list but are presented along with the flow chart for that routine in the flow chart section. Variables that are either input or output variables are defined in Volume II.

Mathematical symbols are presented for each variable where applicable. The common that contains the variable is also shown, along with a definition of the variable and the routine that defines it.

DEFINITION OF INTERNAL FORTRAN SYMBOLS

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
AB(I) I=1,9	-	DPGVC	THE TRANSFORMATION MATRIX FROM THE ATMOSPHERIC RELATIVE SYSTEM TO THE BODY SYSTEM/ IBMTRX
ACOB(I) I=1,625	-	SEARC	THE JACOBIAN OF THE CONSTRAINT VECTOR/ GRAD
AR(I) I=1,15	-	MOTVC	THE TABLE LOOK-UP VALUE OF EXIT AREA FOR ENGINE I/ PROP
CNP1(I) I=1,40	-	DYTEM	CORRECTED SOLUTION IN THE FOURTH ORDER PREDICTOR-CORRECTOR INTEGRATION FORMULA/ FOPC
CTHAT	-	SEARC	COSINE OF CONEPS(1)/ TEST
CYCF	-	CYCVC	CYCLING FLAG SUCH THAT IF =0, THIS IS A DERIVATIVE PASS WITH DELT=0/ CYCXMI, CYCXM2
D(I) I=1,3	-	LOCAL	THE ELEMENTS OF THE D VECTOR USED TO COMPUTE THE QUATERNION RATES/ GUID2
DELT	-	CYCVC	CURRENT INTEGRATION STEP SIZE/ CYCXM
DG(I) I=1,25	-	SEARC	THE DIFFERENCE BETWEEN THE COST GRADIENT ON TWO SUCCESSIVE ITER-ATIONS/ DGM
DFVLH(I) I=1,3	-	MOTVC	THE HISTORY VALUES OF THE FUNCTIONAL INEQUALITY CONSTRAINT DERIVATIVES MOTION
DMI(I) I=1,15	-	MOTVC	THE TABLE LOOK-UP VALUE OF FLOWRATE FOR ENGINE I/ PROP

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
DPR	-	SERVC	DEGREES PER RADIAN CONVERSION FACTOR/ BLKDAT
DTIME	-	CYCVC	THE DERIVATIVE OF TIME, I.E., 1.0
DTO	-	CYCVC	THE INTEGRATION STEP SIZE IN THE CURRENT PHASE/ CYCXMI
DTSAV	-	DYTEM	THE LAST INTEGRATION STEP SIZE. USED TO CHECK FOR CHANGES IN STEP SIZE WHEN USING THE PREDICTOR-CORRECTOR FORMULA/ DYNS2
DXP(I)	-	LOCAL	THE THRUST MOMENT ARM OF ENGINE I
DYP(I)			MASSP
DZP(I) I=1,15			
DXR(I) I=1,3	-	LOCAL	THE AERODYNAMIC MOMENT ARM/ MASSP
DYNTL(I) I=1,223	-	DYNTL	AN ARRAY WHICH CONTAINS THE INTEGRALS, DERIVATIVES, AND A FLAG FOR EACH VARIABLE TO INDICATE WHETHER IT IS TO BE INTEGRATED/ BLKDAT
EA(I) I=1,25	-	SEARC	THE ERROR VECTOR FOR THE ACTIVE CONSTRAINTS/ PGM
EMAP(I) I=1,625	-	SEARC	THE TRANSFORMATION THAT DETERMINES THE DIRECTION OF SEARCH FOR CONSTRAINT SATISFACTION/ UPDATS
ENOIS	-	CYCVC	A SMALL NUMBER USED AS A TOLERANCE TEST
ENOM(I) I=1,25	-	SEARC	THE NOMINAL TARGET ERRORS/ NOMINL

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
EPSA(I) I=1,4	-	SEARC	THE STEPSIZE CONTROL FOR THE ONE-DIMENSIONAL MINIMIZATION ROUTINE, I.E., UPPER AND LOWER BOUNDS AND CURVEFIT ERROR TOLERANCES/ FGAMA, TRYIT1, TRYIT2
ESN	-	TGOVC	THE CURRENT EVENT SEQUENCE NUMBER DINPT
ESNPRT	-	INFVC	THE CURRENT EVENT SEQUENCE NUMBER FOR PRINTOUT/ INFXMI
EVNT(I) I=1,8	-	REDAT	THE EVENT SEQUENCE NUMBER AND CRITERIA ARRAY/ READAT
EVTF	-	PHZVC	EVENT FLAG/ PHZXMI =0, NOT AN EVENT =1, ON MINUS SIDE OF AN EVENT =2, ON PLUS SIDE OF AN EVENT
EXTRAP	-	INFVC	THE VALUE OF TIME WHEN THE LAST PROFIL TIME SLICE WAS WRITTEN. USED TO COMPUTE THE PROFIL WRITE INTERVAL/ INFXM
FPP5	-	SERVC	FLOATING POINT NUMBER = .5
FP00 . . . FP15	-	SERVC	FLOATING POINT NUMBERS 0,1,...,15
FP60	-	SERVC	FLOATING POINT 60
FP90	-	SERVC	FLOATING POINT 90
FP180	-	SERVC	FLOATING POINT 180
FP270	-	SERVC	FLOATING POINT 270

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
FP360	-	SERVC	FLOATING POINT 360
FUXN(I) I=1,10	-	TGOVC	THE HISTORY VALUES OF THE CRITERIA VARIAPLES CURRENTLY BEING MONITORED/ TGOEN
GAMASS	-	SEARC	THE STEPSIZE FOR THE NON-UNITIZED DIRECTION OF SEARCH/ TRYIT1
GAMAX	-	SEARC	THE MAXIMUM STEPSIZE ALLOWED CONSIDERING THE INACTIVE INEQUALITY CONSTRAINTS, AND THE MAXIMUM ALLOWED PERCENTAGE CHANGE IN THE CONTROL PARAMETERS/ TRYIT1
GB(I) I=1,9	-	DPGVC	THE TRANSFORMATION MATRIX FROM THE GEOGRAPHIC SYSTEM TO THE BODY SYSTEM/ IBMTRX
GCON(I) I=1,3	-	LOCAL	CONSTANTS USED BY GRAV IN COMPUTING THE GRAVITATIONAL POTENTIAL/ MOTIAL
GP(I) I=1,25	-	SEARC	THE COST GRADIENT ON THE PREVIOUS ITERATION/ PGM, DGM, DGMP2
GSQOLD	-	SEARC	THE SQUARE OF THE GRADIENT MAGNITUDE ON THE PREVIOUS ITERATION/ CGM
HDG(I) I=1,25	-	SEARC	THE ESTIMATE OF THE HESSIAN MATRIX GENERATED IN THE DAVIDON ALGORITHM/ DGM
HEADER(I) I=1,10	-	INFIC	THE TITLE TO BE PRINTED OUT AT THE TOP OF EACH PAGE/ INFXMI

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
HEATC	-	LOCAL	CONSTANT USED IN AERO HEATING CALCULATION/ MOTIAL
HESS(I) I=1,325	-	SEARC	THE APPROXIMATION TO THE HESSIAN MATRIX GENERATED IN THE DAVIDON ALGORITHM/ DGM, DGMP2, PGM
I,J,K,L,M	-	SERVC	INTEGER VARIABLES FOR TEMPORARY USE
IA(I) I=1,9	-	DPGVC	TRANSFORMATION MATRIX FROM PLANET CENTERED INERTIAL TO ATMOSPHERIC RELATIVE FRAME/ IBMTRX
IACS(I) I=1,25	-	SEARC	THE SAVED VALUES OF THE SUBSCRIPTS OF THE ACTIVE CONSTRAINTS/ TRYIT1
IBKT(I) I=1,24000	-	BLANK	A DATA BUFFER WHICH CONTAINS THE EVENT CRITERIA AND THE TABLE INPUT DATA/ READAT, DINPT
IBTC(I) I=1,25	-	OVRLY25	A COMBINATION OF ACTIVE CONSTRAINT INDICES USED TO DETERMINE IF SOME CONSTRAINTS CAN BE MADE INACTIVE/ DROP
ICASE	-	MULTRC	CURRENT CASE (PROBLEM) NUMBER/ READAT
ICD	-	OVRLY25	INDEX OF THE ACTIVE CONSTRAINT WHICH WAS DROPPED/ REVISE
ICGM	-	SEARC	A FLAG TO INDICATE THAT INITIALIZATION MUST BE DONE IN THE CONJUGATE GRADIENT ROUTINE (CGM) IF ICGM IS NON-ZERO/ DELTU
IDAV	-	SEARC	NOT USED.

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
IDENT(I) I=1,9	-	SERVC	THE IDENTITY MATRIX (3X3)/ BLKDAT
IDTAB(I) I=1,5	-	CYCVC	ADDRESSES OF THE TABLES THAT ARE TO BE USED IN COMPUTING THE INTEGRATION STEP SIZE DURING THE CURRENT PHASE/ CYCXM, DTMDL
IDX	-	REDAT	INDEX USED IN READAT TO MERGE MULTIPLE RUN DATA/ READAT
IENT	-	REDAT	CURRENT ESN BEING MATCHED DURING GENERATION OF MULTIPLE RUN DATA/ READAT
IERRX	-	REDAT	USED TO SAVE FATAL ERROR FLAG BEFORE EVALUATING NON-FATAL ERRORS/ READAT
IESN	-	PHZVC	THE INITIAL EVENT SEQUENCE NUMBER DINPT
IEVNT(I) I=1,10	-	TGOVC	THE ARRAY OF EVENT LOCATIONS CURRENTLY BEING MONITORED/ TGOEMI
IFLG	-	CYCVC	A FLAG USED TO INDICATE THAT AN EVENT HAS BEEN INITIATED/ CYCXM
IFRST	-	REDAT	A FLAG TO INDICATE THAT THIS IS THE BEGINNING OF A PROBLEM/ READAT
IG(I) I=1,9	-	DPGVC	THE TRANSFORMATION MATRIX FROM THE PLANET CENTERED INERTIAL TO THE GEOGRAPHIC SYSTEM/ MOTION
IGEN(I) I=1,2500	-	GENRL	A DATA BUFFER WHICH CONTAINS THE GENERAL INPUT DATA/ READAT, DINPT

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
IHADIT	-	SEARC	A FLAG WHICH IS SET TO 1 IF THE PROGRAM COULD NOT GET TARGETED ON THE LAST OPTIMIZATION STEP/ TRYIT TRYIT2
II	-	REDAT	USED AS INDEX AND COUNTER DURING DATA PROCESSING IN READAT/ READAT
IL(I) I=1,9	-	DPGVC	THE TRANSFORMATION MATRIX FROM THE PLANET CENTERED INERTIAL TO THE LAUNCH INERTIAL SYSTEM/ MOTIAL
IMAX	-	SEARC	THE LAST PHASE NUMBER OF DEPPH(I) AND OPTPH TO OCCUR IN TIME/ MINMYS
IMIN	-	SEARC	THE FIRST PHASE NUMBER OF DEPPH(I AND OPTPH TO OCCUR IN TIME/ MINMYS
IMLT	-	READAT	POINTS TO THE BEGINNING OF LABELED COMMON MNMMLT IN INV/READAT
IMULT	-	MULTRC	SIZE OF GENERAL DATA RECORD SAVED FOR MULTIPLE RUNS/ READAT
IN	-	MULTRC	CURRENT INPUT AND OUPUT FILE FOR
IO			MULTIPLE RUN - FLIP FLOPS BETWEEN 3 AND 4/ READAT
IND(I) I=1,25	-	OVRLY25	COMBINATION OF CONSTRAINTS SELECTED BASED ON COMBINATORIAL PROCEDURES/ COMBINE
INFF	-	INFVC	A PRINT FLAG WHICH FORCES A PRINT WHEN SET NONZERO/ PHZXM
INIT	-	DYTEM	A COUNTER IN PREDICTOR-CORRECTOR TO START THE ALGORITHM/ DYNXB, DYNS2

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
INSRT	-	READAT	FLAG USED TO INDICATE IF A TABLE HAS BEEN INSERTED FOR THE CURRENT PHASE BEING PROCESSED IN MULTIPLE RUN PORTION OF READAT/ READAT
INTRBL	-	SEARC	A FLAG TO INDICATE THAT THE PROGRAM COULD NOT GET TARGETED ON THE CURRENT OPTIMIZATION STEP/ TRYIT2, TEST
INTRYI	-	SEARC	A FLAG TO INDICATE THAT OVERLAY (2,5) IS TO BE CALLED WHEN SET NON-ZERO/ TRYIT1
IOPT	-	SEARC	A FLAG WHICH IS SET NON-ZERO TO INDICATE THAT THERE WERE NO PREVIOUS OPTIMIZATION STEPS/ TRYIT1
IPRT	-	GENIC	A FLAG WHICH SUPPRESSES THE TRAJECTORY PRINTOUT WHEN SET TO ZERO/ MINMYS
IPRNTB	-	INFVC	THE NUMBER OF FULL PRINT LINES IN THE PRINT BLOCK/ INFXM
IPRNTR	-	INFVC	THE NUMBER OF REMAINING PRINT VARIABLES IN THE LAST LINE THAT IS NOT FULL/ INFXM
IRANGE	-	AUXVC	A FLAG TO INDICATE THAT THE RANGE CALCULATION HAS BEEN INITIALIZED/ AUXFMI
IREVDT	-	GENIC	REVISION DATE PRINTED AT THE BEGINNING OF EACH RUN IN READAT/ BLKDAT

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
IRUNF	-	READAT	FLAG USED TO INDICATE IF CURRENT CASE IS TO BE BYPASSED. IRUNF=0 IMPLIFS CASE WILL BE BYPASSED/ READAT
IR1(1)	-	SERVC	A TEMPORARY REUSABLE ARRAY
ISCORE(I) I=1,2	-	MULTRC	WORDS IN WHICH SUCCESS OR FAILURE OF RUNS ARE PACKED/ RSCORE
ISFLG	-	SEARC	FLAG WHICH INDICATES THAT THIS IS A RESTART ITERATION FOR THE DAVIDON OPTION/ RSEARCH
ISTART	-	SEARC	A FLAG TO INDICATE THAT THE DAVIDON ALGORITHM IS TO BE RE-STARTED IF SET NON-ZERO/ TRYIT1, PGM
ISTC(2) I=1,25	-	OVRLY25	INDICES OF THE SAVED RIGHT CONSTRAINTS/ DROP
ISTOP(2) I=1,4	-	SEARC	AN ARRAY TO INDICATE HOW THE CURRENT PROBLEM TERMINATED/ NOMINAL, TEST ISTOP(1)=77B, PROBLEM SOLVED ISTOP(2)=77B, ITERATION LIMIT ISTOP(3)=77B, NO CHANGE IN STATE ISTOP(4)=77B, TIME LIMIT
ISZBLK	-	MULTRC	SIZE OF SEARCH DATA RECORD SAVED FOR MULTIPLE RUNS/ READAT
ISZEV	-	TGOVC	THE NUMBER OF EVENTS BEING MONITORED/ TGOEMI
ISIZ	-	READAT	THE SIZE OF THE TABLE CURRENTLY BEING STORED IN IBKT/ READAT

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
ISV	-	MOTVC	A POINTER TO THE LAST POINT USED IN THE ATMOSPHERE TABLE LOOK-UP/ ATMOS2
ITC(2) I=1,25	-	SEARC	THE INDICES OF THE TIGHT INEQUALITY CONSTRAINTS/ NOMINL, REVISE, DROP
ITERF	-	SEARC	A FLAG TO INDICATE THE TYPE OF ITERATION STEP/ MINMYS =0, TARGETING ONLY =1, OPTIMIZATION ONLY =2, TARGETING AND OPTIMIZATION
IVSZ	-	SERVC	THE SIZE OF THE COMPUTATIONAL COMMON REGION (END-IV 1)/ REDAT
IWTFLG	-	SEARC	A FLAG TO INDICATE THAT THE DEPENDENT AND INDEPENDENT VARIABLES ARE WEIGHTED OR UNWEIGHTED/ WUCAL
IXBKT	-	READAT	THE NUMBER OF CELLS OCCUPIED BY THE CONTENTS OF IBKT/ READAT
IXXH(2) I=1,9	-	LOCAL	THE HISTORY (SAVED) VALUE OF THE INERTIA MATRIX. / MASSP, RMOTI
IXE	-	READAT	A POINTER TO THE EVENT IN IBKT/ READAT
IXEVG	-	READAT	A POINTER USED IN THE GENERAL DATA ARRAY (IGEN)/ READAT
IXEVT	-	READAT	A POINTER USED IN THE TABLE DATA ARRAY (IBKT)/ READAT
IXEVT	-	TGOVC	AN INDEX ON IEVNT/ TGOEMI

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
IXG	-	READAT	POINTS TO BEGINNING OF CURRENT PHASE IN NEW CASE DATA SET/ READAT
IXGEN	-	READAT	THE NUMBER OF CELLS OCCUPIED BY THE CONTENTS OF IGEN/ READAT
IXT	-	READAT	AN INDEX USED TO PACK THE TABLES INTO IBKT/ READAT
IX1	-	READAT	INDICES USED TO BUILD THE DATA
IX2			BUFFERS/ READAT
IXTBL	-	READAT	THE CORE LOCATION OF THE TABLE BEING INSERTED INTO IBKT/ READAT
JAC(2) I=1,26	-	SEARC	THE INDICES OF THE ACTIVE CONSTRAINTS/ TRYIT1
JMLT	-	READAT	POINTS TO END OF LABELED COMMON HOLINC IN IV/ READAT
JMULT	-	MULTRC	SIZE OF TABLE DATA RECORD SAVED FOR MULTIPLE RUN/ READAT
KREEP	-	SEARC	A FLAG WHICH INDICATES IF THE ITERATION IS PROGRESSING TOWARDS A SOLUTION/ TEST
LB(2) I=1,9	-	OPGVC	THE TRANSFORMATION MATRIX FROM THE LAUNCH INERTIAL TO THE BODY SYSTEM/ IBMTRX
LIMGBK	-	READAT	THE MAXIMUM SIZE OF THE GENERAL DATA ARRAY (IGEN)/ READAT
LIMBKT	-	READAT	THE MAXIMUM SIZE OF THE TABLE DATA ARRAY (IBKT)/ READAT

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
LISTD(I) I=1,40	-	DYTEM	THE ADDRESSES OF THE CURRENT DEPRIVATIVES/ DYSI1, DLOOK
LISTI(I) I=1,40	-	DYTEM	THE ADDRESSES OF THE CURRENT INTEGRALS/ DYSI1, DLOOK
LMBKT	-	READAT	THE CURRENT SIZE OF IBKT/ READAT
LMEVT	-	READAT	THE SIZE OF THE CURRENT EVENT BEING INPUT/ READAT
LNGTH	-	READAT	THE LENGTH OF THE COMPUTATIONAL COMMON AREA/ READAT
LPRNT	-	INFVC	THE LAST PRINT TIME/ INFXMI, INFXM
N	-	DYTEM	THE NUMBER OF INTEGRALS IN THE CURRENT PHASE/ DYSI1, DLOOK
NACS	-	SEARC	THE NUMBER OF ACTIVE CONSTRAINTS INCLUDING THE EQUALITY CON-STRAINTS/ TRYIT1
NAMSVR(I) I=1,51	-	SEARC	THE HOLLERITH NAMES OF THE CONTROL, TARGET, AND OPTIM-IZATION VARIABLES/ READAT
NDISC	-	DYNVC	A FLAG TO SIGNAL THE INTEGRA-TION ALGORITHM THAT THERE IS A DISCONTINUITY/ DYNXA
NEQC	-	SEARC	THE NUMBER OF EQUALITY CON-STRAINTS/ MINMYS
NETF	-	SEARC	A FLAG WHICH IS SET NON-ZERO IF SRCHM=4 AND OPT IS NON-ZERO/ MINMYS

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
NFLAG	-	SEARC	A FLAG WHICH IS SET POSITIVE TO INDICATE THAT THE PROBLEM HAS CONVERGED/ TEST
NINDVX	-	READAT	NUMBER OF INDEPENDENT PARAMETERS SAVED FROM LAST RUN/ READAT
NLDADR(I) I=1,25	-	READAT	ADDRESS OF THE CONTROLS SAVED FROM THE LAST SEARCH RUN/ READAT
NLDPH(I) I=1,25	-	READAT	PHASE NUMBERS ASSOCIATED WITH THE CONTROLS SAVED FROM THE LAST SEARCH RUN/ READAT
NOMF	-	SEARC	A FLAG WHICH IS SET NON-ZERO IF THE TRAJECTORY BING RUN IS A NOMINAL TRAJECTORY/ NOMINL
NOO	-	SERVC	FIXED POINT ZERO
NO1	-	SERVC	FIXED POINT NUMBERS 1,...,15
.			
.			
.			
NO15			
NPAGE	-	MULTRC	PAGE COUNTER/ PAGER
NPASS	-	DYNVC	AN INTEGRATION PASS FLAG/ DYNXA, RUK
NPC9	-	LOCAL	INPUT VALUES OF NPC(9), NPC(13),
NPC13			AND NPC(17) THAT ARE REQUIRED FOR
NPC17			PRINTOUT/ PRINTIC
NSTEP	-	SEARC	AN ITERATION COUNTER FOR THE CURRENT PROBLEM/ MINMYS

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
NTC	-	SEARC	THE NUMBER OF TIGHT CONSTRAINTS/ REVISE
NULL	-	SERVC	A VARIABLE USED OT DETECT IF INPUT VARIABLES HAVE BEEN INPUT. THE STORED VALUE OF NULL IS 1HU/ BLKDAT
OLDG2		SEARC	G2MAG FROM THE PREVIOUS INTERATION/ TEST
OLDP1		SEARC	P1NOM AND P2NOM FROM THE PREVIOUS
OLDP2			ITERATION/ TEST
OLDU		SEARC	UMAG FROM THE PREVIOUS ITERATION/ TEST
OMGSLT	-	LOCAL	OMEGA*SIN(LATC)/ MOTIAL, MOTION
ONE	-	MNMMLT	THE NAME OF A CELL WHICH CONTAINS FLOATING POINT ONE/ NOMHOL
PTCGO	-	TGOVC	PERCENT OF NOMINAL INTEGRATION STEP ALLOWED BY TIME-TO-GO LOGIC TO BRACKET THE DESIRED FUNCION VALUE/ TGEON
PCTOLD	-	SEARC	THE MAXIMUM PERCENTAGE CHANGE ALLOWED ON THE PREVIOUS OPTIM-IZATION STEP/ TRYT1
PE(I) I=1,240	-	INFIC	THE CORE ADDRESSES AND NAMES OF THE CURRENT PRINT VARIABLES/ INFXMI, INFXM
PHZF	-	PHZVC	A FLAG TO INDICATE THAT THE TRAJ-ECTORY IS TO BE TERMINATED/ PHZXM =0, DO NOT TERMINATE THE TRAJECTORY NE 0, TERMINATE THE TRAJECTORY

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
PI		SERVC	PI/ BLKDAT
PIF	-	PHZVC	A PROGRAM INITIALIZATION FLAG/ PHZXMI =0, THIS IS THE INITIALIZATION PASS =1, THIS IS NOT THE INITIALIZAITON PASS
PIO2		SERVC	PI OVER TWO/ BLKDAT
PNMCN(I) I=1,40	-	DYTEM	FIRST DIFFERENCE BETWEEN PREDICTED SOLUTION AND CORRECTED SOLUTION USED IN FOURTH ORDER PREDICTOR-CORRECTOR FORMULA/ FOPC
PNPI(I) I=1,40	-	DYTEM	PREDUCTED SOLUTION IN FOURTH ORDER PREDICTOR-CORRECTOR INTEGRATION FORMULA/ FOPC
PROJ(I) I=1,625	-	OVRLY25	THE MATRIX WHICH DETERMINES THE PROJECTED GRADIENT/ UPDATS, REVISE
PINOM	-	SEARC	THE VALUES PF P1 AND P2 ON THE
P2NOM			NOMINAL TRAJECTORY/ MINMYS
P2MIN	-	SEARC	THE LOWER BOUND ON THF WIEGHTED ERROR MAGNITUDE/ DELTU
QLDU(I) I=1,25	-	REDAT	CONTROLS SAVED FROM LAST SEARCH RUN/ READAT
QSREF	-	LOCAL	DYNAMIC PRESSURE TIMES THE AERODYNAMIC REFERENCE AREA/ MOTION
RERP2	-	LOCAL	(RE/RP)**2/ MOTIAL
REVNT	-	REDAT	A VARIABLE USED TO SET A FLAG FOR A ROVING EVENT/ READAT

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
RPD	-	SERVC	RADIANS PER DEGREE CONVERSION FACTOR/ BLKDAT
S(I) I=1,25	-	SEARC	THE VALUE OF DU FORM THE PREVIOUS ITERATION/ CGM
SALPHA	-	LOCAL	THE SINE AND COSINE OF THE ANGLE
CALPHA			OF ATTACK/ GUID1, MOTION
SAVE(I) I=1,70	-	TGOVC	THE VALUES OF THE STATE VARIABLES AT THE LAST INTEGRATION STEP/ TGOEM
SAVIT(I,J) I=1,27	-	SEARC	AN ARRAY IN WHICH THE RESULTS OF THE CURVEFIT OF P1 AND P2 ARE
J=1,5			SAVED/ FGAMA
SBANK	-	LOCAL	THE SINE AND COSINE OF THE ANGLE
CBANK			BANK/ GUI1, MOTION
SBETA	-	LOCAL	THE SINE AND COSINE OF THE ANGLE
CBETA			SIDESLIP/ GUID1, MOTION
SFCT(I) I=1,15	-	LOCAL	THE TABLE LOOK-UP VALUE OF THE SPECIFIC FUEL CONSUMPTION FOR ENGINE I/ PROP
SGAMA	-	LOCAL	THE SINE AND COSINE OF THE PATH
CGAMA			ANGLE RELATIVE TO THE ATMISPHERE/ GUID1, GUI2
SIDEAL	-	AUXVC	THE SAVED VALUE OF VIDEAL USED TO COMPUTE THE REQUIRED VELOCITY MARGIN/ AUXFMI

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
SINDPH(I) I=1,25	-	SEARC	THE VALUES OF INDPH(I) WHICH ARE SAVED IN READAT FOR PRNTIC/ READAT
SAZVA	-	LOCAL	THE SINE AND COSINE OF THE AZIMUTH
CAZVA			ANGLE RELATIVE TO THE ATMOSPHERE/ GUI1, GUID2
SAZREF	-	LOCAL	THF SINE AND CONSINE OF THE REFERENCE
CAZREF			AZIMUTH ANGLE USED IN THE RANGE CALCULATIONS/ AUXFMI
SGCLAT	-	LOCAL	THE SINE, COSINE, AND TANGENT OF
CGCLAT			THE GFOCENTRIC LATITUDE OF THE
TGCLAT			VEHICLE/ MOTIAL, MOTION
SLATRF	-	LOCAL	THE SINE AND COSINE OF THE
CLATRF			REFERFNCE LATITUDE USED IN THE RANGE CALCULATIONS/ AUXFMI
SLG	-	LOCAL	SINE AND COSINF OF GEODETIC
CLG			LATITUDF/ MOTIAL
SLONG	-	LOCAL	SINF AND COSINE OF INERTIAL
CLONG			LONGITUDE/ MOTIAL, MOTION
SOLVED	-	ULTRC	FLAG SET TO INDICATE HOW THF LAST PROBLEM TERMINATED/ SCORE
SSTI(I) I=1,625	-	OVRLY25	A MATRIX USED IN COMPUTING THE PROJECTING MATRIX AND IS EQUAL TO (S*S PRIME)INVEPSE/ UPDATS, PGM

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
STEMP(I) I=1,25	-	SERVC	AN ARRAY USED FOR TEMPORARY STORAGE
STEP(I) I=1,6	-	SEARC	THE TRIAL STEP LENGTH FOR EACH TRIAL STEP/ GENMIN
SRMINP(I) I=1,2	-	SEARC	THE MINIMUM STEP SIZE DECREASE FROM THE PREVIOUS STEP WHEN GENERATING THE CURVEFIT/ GENMIN
STPMAX	-	SEARC	THE LENGTH OF THE STEP IN THE DIRECTION OF SEARCH REQUIRED TO REACH THE BOUNDARY FOR THE NEAREST INEQUALITY CONTRAINTS / TRYIT1, TRYIT2
TABLE(I) I=1,2500	-	REDAT	THE CURRENT TABLE BEING INPUT. USED TRANSFER THE TABLE FROM INPUT TO THE STORAGE ARRAY IBKT/ RTAB, READAT
TEMP(I) I=1,50	-	SERVC	AN ARRAY USED FOR TEMPORARY STORAGE
TGO	-	TGOVC	THE TIME TO GO TO THE NEXT EVENT/ TGOEM
TGRAD	-	SEARC	TIME REQUIRED TO COMPUTE ALL TRAJECTORY SENSITIVITIES/ MINMYS
TI(I) I=1,15	-	MOTVC	THE VALUE OF NET THRUST FOR ENGINE I/ PROP
TIMIN	-	SEARC	THE TIME AT WHICH THE EARLIEST PHASE IN INDPH(I) OCCURS/ NOMINL, SAVIC
TIMX	-	TGOVC	THE LAST VALUE OF TIME/ TGOEM
TJD(I) I=1,15	-	LOCAL	THE TABLE LOCK-UP VALUE OF THRUST FOR ENGINE I/ PROP

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
			CYCXM
TREF	-	CYCVC	A TIME REFERENCE USED TO COMPUTE THE NEXT STEP SIZE (DELT)/ CYCXMI
TREFP	-	DYNVC	THE TIME REFERENCED TO THE LAST PRIMARY EVENT, CYCXMI, TGEOMI
TREFS	-	DYNVC	THE TIME REFERENCED TO THE LAST SECONDARY EVENT/ CYCXMI, TGOEMI
TTRIAL	-	SEARC	TIME REQUIRED FOR A TRIAL STEP IN THE ONE DIMENSIONAL MINIMIZATION ROUTINE/ FGAMA
TWOPI		SERVC	TWO TIMES PI/ BLKDAT
WECONH	-	LOCAL	THE HISTORY (SAVED) VALUE OF WECON/ WGTINI
WGTO	-	LOCAL	THE CALCULATED VALUE OF INITIAL STAGE WEIGHT/ WGTINI
WJETTO	-	LOCAL	THE SAVED VALUE OF JETTISON WEIGHT FOR PRINTOUT PURPOSES/ WGTINI
WPROPO	-	LOCAL	INITIAL STAGE PROPELLANT WEIGHT/ WTGINI, PROP
WPUSHDH	-	LOCAL	PROPELLANT CONSUMED UP TO AND INCLUDING LAST PHASE/ WTGINI
XINF	-	SERVC	FLOATING POINT INFINITY (10.E10)/ BLKDAT
ZIXAV(I) I=1,3	-	AUXVC	THE SAVED VALUES OF THE VEHICLE POSITION VECTOR RELATIVE TO THE EARTH FOR USE IN COMPUTIING DPRNG1 AND DPRNG2/ AUXFMI
XYOME(I) I=1,3	-	DPGVC	CONSTANTS USED TO COMPUTE THROTTLE COMMAND WHEN USING GENERALIZED LINEAR COMMNADS/ MOTIAL, PROP

DEFINITION OF INTERNAL FORTRAN SYMBOLS (CONTD)

FORTRAN SYBMOL	MATH SYMBOL	COMMON	DEFINITION/SUBROUTINE
YES(I) I=1,4	-	SEARC	THE VALUES OF THE POLYNOMIALS GENERATED DURING THE ONE DIMENSIONAL MINIMIZATION AT THEIR PREDICTED MINIMUMS/ TRYIT1
YP(I,J) I=1,4 J=1,40	-	DYTEM	BACK DERIVATIVES USED IN FOURTH ORDER PREDICTOR-CORRECTOR FORMULA/ FOPC
YPVAV(I) I=1,40	-	DYTEM	INITIAL VALUES OF INTEGRALS USED IN PREDICTOR-CORRECTOR FORMULA/ FOPC
Y1(I)	-	DYTEM	WORKING STORAGE USED BY THE RUNGE-
Y2(I) I=1,40			KUTTA INTEGRATION ALGORITHM/ RUK
Z1TRY(I) I=1,6	-	SEARC	SAVED VALUES OF P1TRY(I)/ TRYIT1
Z2TRY(I) I=1,6	-	SEARC	SAVED VALUES OF P2TRY(I)/ TRYIT1
ZMAG	-	SEARC	SAVED VALUE OF UMAG/ TRYIT1
ZMAX	-	SEARC	SAVED VALUE OF STPMAX/ TRYIT1
ZP1DS	-	SEARC	SAVED VALUE OF DP1DS/ TRYIT1
ZP2DS	-	SEARC	SAVED VALUE OF DP2DS/ TRYIT1
ZSTEP(I) I=1,6	-	SEARC	SAVED VALUES OF THE TRIAL STEP SIZES/ TRYIT1
ZYES(I) I=1,5	-	SEARC	SAVED VALUFS OF YES(I)/ TRYIT1

VI. POST SUBROUTINE INDEX

SUBROUTINE NAME	OVERLAY	OCTAL SIZE
ABT	(2,5)	21
ABXT	(0,0)	2
AERO	(2,3)	756
AEROHI	(2,3)	207
AERO4	(2,3)	124
AIREM	(2,3)	15
ANGLE2	(2,3)	106
ANGV2	(2,3)	40
ANPART	(2,0)	3
ANTANH	(2,3)	16
ATAN3	(2,0)	7
ATMOS	(2,0)	26
ATMOS1	(2,0)	35
ATMOS2	(2,0)	237
ATMOS3	(2,0)	427
AUTCPM	(2,3)	20
AUTP1	(2,3)	511
AUTP2	(2,3)	1076
AUXEM	(2,3)	365
AUXEMI	(2,2)	167
BACKOI	(2,0)	36
BACKOP	(2,0)	41
BLKDAT	(0,0)	0
BTW	(2,5)	17
BUCKET	(2,0)	105
CALE	(2,0)	61
CALSPE	(2,3)	20
CGM	(2,5)	27
CLGM	(2,3)	7
CLSPFL	(2,0)	74
COMBIN	(2,5)	44
CONIC	(2,3)	505
CONTM	(2,3)	37
CONVO	(2,3)	304
CONVO1	(2,3)	24
CUBMIN	(2,0)	103
CYCXM1	(2,3)	41
CYCXM2	(2,3)	202
CYCXMI	(2,2)	104
DATA	(2,0)	3
DELTU	(2,5)	33
DERIV	(2,3)	16
DERVI	(2,2)	10
DGAMLA	(2,3)	131
DGM	(2,5)	130

POST SUBROUTINE INDEX (CONTD)

SUBROUTINE NAME	OVERLAY	OCTAL SIZE
DGMP2	(2,5)	30
DICT	(1,0)	3
DIGDIF	(2,0)	37
DINPT	(2,1)	77
DLOCK	(2,2)	51
DPRNG	(2,3)	155
DQUAT	(2,3)	24
DROP	(2,5)	273
DTMOL	(2,3)	111
DUCAL	(2,5)	6
DWINDS	(2,3)	35
DYNS2	(2,3)	14
DYNXA	(2,2)	7
DYNXM	(2,3)	17
DYNXMI	(2,2)	12
DYSI1	(2,2)	32
ERROR	(0,0)	47
EXPN	(1,0)	52
FANDE	(2,0)	14
FGAMA	(2,0)	160
FOPC	(2,3)	176
FOPMIN	(2,0)	142
GABDD	(2,5)	47
GAMLAM	(2,0)	65
GCNTRL	(2,3)	3
GEMNIN	(2,0)	505
GENTAB	(2,0)	271
GGUID	(2,3)	17
GGUID1	(2,3)	25
GGUID2	(2,3)	1403
GMAG	(2,5)	44
GNAV	(2,3)	3
GRAD	(2,0)	442
GRAV	(2,3)	65
GSENSR	(2,3)	76
HINGEM	(2,3)	41
IBMTRX	(2,0)	51
IBMTXI	(2,2)	203
ICKTAB	(2,2)	21
INFXM	(2,3)	206
INFXMI	(2,2)	165
INPUTX	(1,0)	647
INTGRL	(2,2)	25
INVM	(2,3)(2,5)(2,6)	415
IRTBR	(2,3)	32

POST SUBROUTINE INDEX (CONTD)

SUBROUTINE NAME	OVERLAY	OCTAL SIZE
ITERO	(2,6)	1606
LDRL	(2,3)	50
MASSP	(2,3)	167
MASTER	(0,0)	7,343/11,431
MATFY	(2,0)	20
MINMYI	(2,4)	117
MINMYS	(2,0)	62
MINITR	(2,3)	24
MOTAB	(2,3)	157
MOTIAL	(2,2)	1303
MOTION	(2,3)	273
MTRXM	(2,0)	41
MTRXT	(2,3)	41
MTRXTV	(2,3)(2,2)	20
MTRXV	(2,3)(2,2)	20
NOMPOL	(1,0)	276
NOMINL	(2,0)	151
OLGM	(2,3)	53
ORBTR	(2,2)	131
OUTIT	(2,6)	6
PAGER	(0,0)	46
PAO	(2,0)	65
PAXCAL	(2,2)	206
PBLOCK	(2,3)	275
PGM	(2,5)	210
PHZXM	(2,3)	210
PHZXMI	(2,2)	22
POLY	(2,0)	6
PPT	(2,6)	21
PQI	(2,0)	35
PRNTIC	(2,2)	3617
PROP	(2,3)	303
PTOIC	(2,4)	532
QMULT	(2,2)	33
QRTATE	(2,2)	65
QUADMN	(2,0)	46
QUAT1	(2,2)	54
QUAT2	(2,2)	24
QUAT3	(2,2)	45
READAT	(1,0)	2606
REVDAT	(0,0)	3
REVISE	(2,5)	117
RGENDA	(1,0)	1020
RMOTI	(2,2)	206
RMOTM	(2,3)	135

POST SUBROUTINE INDEX (CONTD)

SUBROUTINE NAME	OVERLAY	OCTAL SIZE
RSCORE	(1,0)	52
RSEARC	(1,0)	214
RTAB	(1,0)	102
RTBLML	(1,0)	1233
RUK	(2,3)	77
RUK2	(2,3)	40
SAVIC	(2,0)	54
SCORE	(1,0)	20
SDM	(2,5)	10
SERCH	(1,0)	32
SETIC	(2,0)	122
SETIV	(2,0)	36
SHRINK	(1,0)	50
SP	(2,0)	10
SREL	(1,0)	43
SYMATS	(2,5)	113
SYMSTA	(2,5)	113
TEST	(2,0)	114
TGOEM	(2,3)	307
TGOEMI	(2,2)	275
THRM	(2,0)	73
THPCSM	(2,0)	76
TMOTM	(2,3)	37
TOIN	(2,4)	7
TPOSM	(2,0)	46
TRAJ	(2,0)	225
TRYIT1	(2,0)	275
TRYIT2	(2,0)	153
TSPXM	(2,0)	33
UNITOU	(2,0)	32
UPDATS	(2,5)	2546
UPNOM	(2,0)	12
VCROSS	(2,3)	12
VDOT	(2,3)	10
VUNIT	(2,3)	20
WGTINI	(2,2)	73
WINDS	(2,0)	60
WUCAL	(2,4)	26
WRNGE1	(2,3)	52
ZEROM	(2,5)	14
ZLAGF	(2,3)	110
ZWASHF	(2,3)	30